



# Proposition d'une nouvelle méthode d'accès déterministe pour un réseau personnel sans fil à fortes contraintes temporelles

Adrien van den Bossche

## ► To cite this version:

Adrien van den Bossche. Proposition d'une nouvelle méthode d'accès déterministe pour un réseau personnel sans fil à fortes contraintes temporelles. Réseaux et télécommunications [cs.NI]. Université Toulouse le Mirail - Toulouse II, 2007. Français. NNT : . tel-00254469

**HAL Id: tel-00254469**

**<https://theses.hal.science/tel-00254469>**

Submitted on 13 Feb 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**LABORATOIRE LATTIS EA 4155**

**Groupe SCSF**

**Systèmes Communicants Sans Fil**

## **MEMOIRE de THESE**

**DOCTORAT de L'UNIVERSITÉ de TOULOUSE II**

**Ecole Doctorale Systèmes**

**Spécialité : Génie Informatique, Automatique et Traitement du Signal**

**Présenté par Monsieur Adrien VAN DEN BOSSCHE**

**Ingénieur ESEO**

**Maître ès Sciences Université Paris VI**

**« Proposition d'une nouvelle méthode d'accès déterministe pour un réseau  
personnel sans fil à fortes contraintes temporelles »**

**Soutenue le 6 juillet 2007 devant le jury composé de :**

Rapporteurs :	F. LEPAGE D. SIMPLOT-RYL	Professeur (61) à l'Université Henri Poincaré Nancy I Professeur (27) à l'Université des Sciences et Technologies de Lille I
Examineurs :	D. FOURNIER-PRUNARET M. MISSON	Professeur (61) à l'INSA de Toulouse Professeur (27) à l'Université de Clermont I
Directeur :	T. VAL	Professeur (61) à l'Université de Toulouse II
Co-encadrant :	E. CAMPO	Professeur (63) à l'Université de Toulouse II
Invité :	C. ZARADER	Directeur technique chez Freescale Toulouse

Léonard de Vinci

# Remerciements

Je remercie vivement Monsieur le Professeur *Thierry Val*, directeur adjoint du laboratoire LATTIS, responsable du groupe de recherche SCSF et directeur de ma thèse, d'avoir accepté de prendre la direction de mes travaux. Je tiens à le remercier très particulièrement pour l'excellence de son accompagnement, ainsi que pour la confiance et la grande autonomie qu'il m'a accordées pendant ces trois années, tout en étant très disponible dès que j'en exprimais le besoin, et ce, malgré ses nombreuses contraintes. Je lui suis tout particulièrement reconnaissant pour son aide active quant à mon intégration dans le milieu de la recherche et pour les responsabilités enrichissantes qu'il a accepté de me confier pendant le doctorat.

J'exprime ma gratitude à Monsieur le Professeur *Eric Campo*, co-encadrant de ma thèse, pour avoir accepté de codiriger les travaux. Je tiens à le remercier vivement pour ses nombreux et pertinents conseils durant toute la durée de mon doctorat, mais aussi pour sa disponibilité et sa grande rigueur, y compris dans les moments où le temps nous était compté, et ce, malgré ses multiples engagements.

Je tiens également à exprimer ici toute ma reconnaissance à Monsieur le Professeur *Jean-Jacques Mercier*, précédant directeur du laboratoire, pour ses généreux et constants appuis, ses nombreux encouragements ainsi que pour la richesse de ses conseils. Je le remercie également de m'avoir accueilli dans le laboratoire et témoigné une grande confiance, et ce, depuis mon premier stage dans l'équipe.

Je remercie Messieurs les Professeurs *Francis Lepage* et *David Simplot-Ryl* pour l'intérêt qu'ils ont porté à mes travaux en ayant accepté de les rapporter et de participer au jury de la thèse. Je remercie également Madame le Professeur *Danielle Fournier-Prunaret*, directrice du LATTIS, pour sa participation au jury, ainsi que Monsieur le Professeur *Michel Misson*, pour son attention, ses conseils judicieux et sa présence au jury. Je remercie également Monsieur *Cyril Zarader*, Directeur Technique chez FREESCALE Toulouse, pour sa disponibilité ainsi que son aide technique durant toute la durée du doctorat. Merci également à lui d'avoir accepté l'invitation du jury.

Je veux également remercier tous les membres de l'équipe : permanents – *Gilles, Fabrice, Jean-François, Laurent* et *Laurence*, doctorants et post-doctorants – *Salim, Céline, Ahcène, Tarik, Nicolas* et *Jackson*, ainsi que les stagiaires, qui ont tous largement contribué à l'excellente atmosphère du laboratoire. Je tiens également à remercier ici les membres des équipes du groupe L2I. Merci à tous et à chacun, pour l'aide et leurs encouragements qui ont contribué à la réussite des travaux.

Je remercie également chaleureusement le personnel de l'IUT de Blagnac : enseignants, techniciens, personnel administratif et collègues, pour l'aide et l'intégration qu'ils m'ont offertes sans lesquelles ces trois années n'auraient pas été aussi enrichissantes.

Je tiens également à exprimer ma gratitude à l'Ecole Doctorale Systèmes de Toulouse de m'avoir accordé une bourse pour financer mes travaux de thèse. Merci également de m'avoir permis de participer à l'organisation du congrès des Doctorants 2005.

Je remercie affectueusement ma famille – mes parents, grands-parents, frères et sœur, cousins ainsi que mes amis de *La Troupe* et de Toulouse pour l'aide qu'ils m'ont apportée, mais aussi leur soutien et leur patience pendant ces trois années où mon investissement a pesé sur ma disponibilité.

Enfin, je tiens à remercier très affectueusement *Marie* pour sa grande confiance et son soutien malgré la distance qui nous sépare. Merci pour sa patience et sa disponibilité qui nous ont permis de nous retrouver régulièrement malgré mon intense activité au laboratoire ; merci enfin à son affection qui a, elle aussi, largement contribué à la réussite de mes trois années de doctorat.



# Table des matières

<b>1</b>	<b>Les réseaux sans fil et la Qualité de Service</b>	<b>11</b>
1	État de l'art des technologies de réseaux sans fil . . . . .	13
1.1	Classification des technologies réseaux . . . . .	13
1.2	Les réseaux sans fil étendus (WWAN) . . . . .	14
1.3	Les réseaux sans fil métropolitains (WMAN) . . . . .	14
1.3.1	La Boucle Locale Radio (BLR) . . . . .	14
1.3.2	ETSI HiperMAN / IEEE 802.16, WiMAX . . . . .	15
1.3.3	IEEE 802.20, Mobile Broadband Wireless Access (MWBA) . . . . .	16
1.3.4	Le phénomène des réseaux sans fil ruraux et des réseaux sans fil coopérants	16
1.4	Les réseaux locaux sans fil (WLAN) . . . . .	16
1.4.1	ETSI HiperLAN . . . . .	17
1.4.1.1	HiperLAN/1 (HIGH PERFORMANCE LAN, première version) . .	17
1.4.1.2	HiperLAN/2 (HIGH PERFORMANCE LAN, seconde version) . . .	17
1.4.2	IEEE 802.11, WiFi . . . . .	18
1.4.2.1	La norme initiale . . . . .	18
1.4.2.2	Topologies . . . . .	18
1.4.2.3	Les évolutions de 802.11 . . . . .	19
1.5	Les réseaux personnels sans fil (WPAN) . . . . .	21
1.5.1	IrDA . . . . .	21
1.5.2	HomeRF . . . . .	22
1.5.3	IEEE 802.15.1, Bluetooth . . . . .	22
1.5.4	IEEE 802.15.3, Ultra Wide Band (UWB) . . . . .	24
1.5.5	IEEE 802.15.4, ZigBee . . . . .	25
1.6	Les réseaux de capteurs (WSN) . . . . .	25
<b>2</b>	<b>La Qualité de Service dans les réseaux</b> . . . . .	<b>26</b>
2.1	Présentation générale . . . . .	26
2.1.1	Qu'est ce que la Qualité de Service ? . . . . .	26
2.1.2	QoS vs. Surdimensionnement . . . . .	26
2.2	Notions fondamentales et principes de QoS . . . . .	27
2.2.1	La prédictabilité du trafic . . . . .	27
2.2.2	Le contrôle d'admission . . . . .	28
2.2.3	Les degrés de Qualité de Service . . . . .	28
2.3	Quelques exemples de protocoles à QoS . . . . .	29
2.3.1	Asynchronous Transfer Mode (ATM) . . . . .	29
2.3.2	IntServ . . . . .	30
2.3.2.1	Profils de QoS au niveau utilisateur . . . . .	30
2.3.2.2	Le protocole de signalisation RSVP . . . . .	30
2.3.2.3	Conclusion : bénéfices et difficultés liés à IntServ . . . . .	31
2.3.3	DiffServ . . . . .	32
2.4	Conclusion . . . . .	32
<b>3</b>	<b>Problématique de l'accès au médium partagé</b> . . . . .	<b>33</b>
3.1	Accéder au médium partagé : les principes de base . . . . .	33
3.2	Les principales techniques d'accès au médium . . . . .	34
3.2.1	CSMA/CA (IEEE 802.11 DCF) . . . . .	34

---

3.2.2	EY-NPMA (HiperLAN/1)	37
3.3	Introduire de la QoS au niveau accès au médium	39
3.3.1	TDMA Dynamique (HiperLAN/2)	39
3.3.2	Point Coordination Function (IEEE 802.11 PCF)	40
3.3.3	Enhanced DCF et Hybrid CF (IEEE 802.11e)	41
3.3.3.1	Enhanced DCF	42
3.3.3.2	Hybrid CF	42
3.3.3.3	Autres améliorations	43
3.3.4	Bluetooth	43
4	Conclusion	44
<b>2</b>	<b>Présentation des normes IEEE 802.15.4 / ZigBee</b>	<b>51</b>
1	Généralités	53
1.1	Le projet ZigBee	53
1.2	Objectifs et domaine d'application	53
1.3	Consommation énergétique	53
1.4	Implémentations	54
1.5	Topologies	55
1.5.1	Topologie étoile	55
1.5.2	Topologie point à point	55
1.5.3	Topologies plus complexes	55
1.6	Adressage	56
1.7	Valeurs typiques	56
2	Présentation de la pile protocolaire ZigBee	56
2.1	Quelques notions fondamentales	56
2.1.1	Le découpage en différentes couches	57
2.1.2	Le principe de l'encapsulation	57
2.1.3	Protocole d'échange entre deux couches voisines	58
2.1.4	Représentation de la pile protocolaire ZigBee	59
2.1.5	Les interfaces de communication entre couches	59
2.2	La couche Physique	60
2.2.1	Bandes de fréquences et canaux	60
2.2.2	Modulations, étalement de spectre	61
2.2.3	Portée, puissance d'émission et sensibilité du récepteur	61
2.2.4	Le paquet de niveau physique	62
2.2.5	Services rendus	63
2.3	La couche Liaison	63
2.3.1	La sous-couche MAC	63
2.3.1.1	Types d'accès au médium	63
2.3.1.2	Notion de supertrame	65
2.3.2	La sous-couche LLC	66
2.3.3	Structures de trames	67
2.3.3.1	Le champ de contrôle de trame	67
2.3.3.2	Le champ de numéro de séquence	68
2.3.3.3	Le champ d'adressage	68
2.3.3.4	Le champ <i>payload</i>	68
2.3.3.5	Le champ FCS	69
2.4	La couche Réseau	70
2.4.1	Éléments de la topologie du réseau	70
2.4.1.1	Topologie en arbre	70
2.4.1.2	Topologie maillée	72
2.4.2	Architecture de la couche réseau	72
2.4.3	Services rendus	73
2.4.4	Adressage	73
2.4.4.1	Adressage NHLE-based	73
2.4.4.2	Adressage en arbre	73

2.4.5	Principes de base du routage ZigBee . . . . .	74
2.4.5.1	Algorithme de routage à la demande . . . . .	75
2.4.5.2	Algorithme de routage hiérarchique : Tree Routing . . . . .	75
2.4.6	Ordonnancement des beacons dans un réseau avec arborescence . . . . .	76
2.4.7	Structure du paquet de niveau réseau . . . . .	76
3	Identification des failles dans la méthode d'accès proposée par le standard . . . . .	77
3.1	La gestion des slots garantis ne présente pas une garantie absolue . . . . .	77
3.2	Autres problèmes identifiés . . . . .	78
3.3	Conclusion : les améliorations proposées . . . . .	79
<b>3</b>	<b>Vers un WPAN à Qualité de Service pour des transports d'informations à fortes contraintes temporelles</b>	<b>83</b>
1	Une application typique de robotique mobile communicante sans fil . . . . .	85
1.1	Problématique générale : pourquoi introduire des communications sans fil pour des robots ? . . . . .	85
1.2	Les attentes vis-à-vis d'un système de communication adapté . . . . .	86
1.2.1	Un système proposant des garanties sur le plan temporel . . . . .	86
1.2.2	Retransmissions vs. redondance et codes correcteurs d'erreurs . . . . .	86
1.2.3	Taille du réseau . . . . .	87
1.2.4	Topologie du réseau . . . . .	87
1.2.4.1	Communications locales au robot . . . . .	87
1.2.4.2	Communications entre robots . . . . .	87
1.2.5	Débit nécessaire . . . . .	88
1.2.6	Conclusion . . . . .	89
2	La proposition protocolaire . . . . .	90
2.1	Nouvelles fonctionnalités proposées . . . . .	90
2.2	Problématique de la cohabitation de plusieurs étoiles sur un médium commun . . . . .	91
2.3	Étude du protocole mis en œuvre . . . . .	92
2.3.1	Éléments du réseau, liens et portées . . . . .	92
2.3.2	Organisation temporelle de l'accès au médium . . . . .	93
2.3.3	Mécanisme de demande de réservation du médium . . . . .	94
2.3.4	Allocations au préalable : arrivées déterministes dans le réseau . . . . .	94
2.3.5	Gestion des acquittements . . . . .	95
2.3.6	Gestion et diffusion des informations de réservation . . . . .	96
2.3.7	Un exemple de demande de réservation du médium . . . . .	96
2.3.8	Une politique d'accès déterministe par défaut . . . . .	99
2.3.9	Optimisation de l'accès au médium : attribution de GTS simultanés . . . . .	100
2.3.9.1	Le principe . . . . .	100
2.3.9.2	Nécessité d'une négociation protocolaire . . . . .	100
2.3.9.3	Application à notre topologie . . . . .	102
2.4	Implémentation du protocole : intégration dans une pile IEEE 802.15.4 . . . . .	104
3	Conclusion . . . . .	105
<b>4</b>	<b>Modélisation, validation et prototypage du protocole</b>	<b>109</b>
1	Introduction : une démarche de validation multi-outils . . . . .	111
2	Modélisation et validation formelle du protocole . . . . .	112
2.1	L'outil Réseaux de Petri . . . . .	112
2.2	Présentation du modèle proposé . . . . .	112
2.2.1	Algorithme général déduit du protocole . . . . .	112
2.2.2	Un modèle de type <i>Père/Fils</i> . . . . .	114
2.2.3	Le modèle Réseaux de Petri . . . . .	115
2.2.3.1	Étude du processus père . . . . .	116
2.2.3.2	Étude du processus fils . . . . .	116
2.3	Validation formelle du modèle et analyse des résultats . . . . .	116
2.3.1	Présentation de l'outil TINA . . . . .	116
2.3.2	Analyse des résultats . . . . .	117

2.4	Conclusion de l'étude de validation formelle . . . . .	117
3	Simulation : outils et résultats . . . . .	117
3.1	Développement d'un outil de simulation . . . . .	117
3.1.1	Présentation de l'outil . . . . .	118
3.1.1.1	Mécanisme implémenté et simulé . . . . .	118
3.1.1.2	Fonctionnement du simulateur développé . . . . .	119
3.1.1.3	Utilisation du simulateur : descriptif de la topologie et exploitation des données . . . . .	119
3.1.1.4	Paramètres fixés pour la simulation . . . . .	120
3.1.2	Analyse des résultats obtenus . . . . .	121
3.1.2.1	Étude de l'association déterministe d'un coordinateur . . . . .	121
3.1.2.2	Étude de l'association déterministe d'un nœud . . . . .	123
3.2	Les travaux réalisés sous NS2 . . . . .	125
3.2.1	Présentation de l'outil . . . . .	125
3.2.2	Les modèles 802.15.4 / ZigBee existants sous NS2 . . . . .	125
3.2.3	Un résultat obtenu avec NS2 : débit utile dans la CAP avec CSMA/CA . . . . .	126
3.3	Conclusion sur les travaux de simulation . . . . .	127
4	Prototype et métrologie . . . . .	127
4.1	Présentation de l'existant . . . . .	127
4.1.1	Les plateformes IEEE 802.15.4/ZigBee existantes . . . . .	127
4.1.2	Modification du firmware existant . . . . .	129
4.2	Réalisation d'outils de tests et de mesures . . . . .	130
4.3	Conception et réalisation d'une pile prototype . . . . .	131
4.3.1	Évaluation des performances de la pile prototypée . . . . .	131
4.3.1.1	Évaluation de la sensibilité du récepteur . . . . .	131
4.3.1.2	Évaluation du temps de traitement d'un paquet . . . . .	132
4.3.1.3	Débit utile maximum . . . . .	133
4.3.1.4	Évaluation de la qualité de la synchronisation entre un coordinateur et ses nœuds . . . . .	134
4.3.1.5	Conclusion . . . . .	135
4.3.2	Validation de notre proposition par prototypage . . . . .	136
4.3.2.1	Évaluation de la qualité de la synchronisation entre le super-coordinateur et ses nœuds . . . . .	136
4.3.2.2	Capacité maximale d'un slot . . . . .	137
4.3.2.3	Validation du concept de SGTS . . . . .	139
4.3.3	Conclusion sur les mesures réalisées sur le prototype . . . . .	142
4.4	Applications de démonstration . . . . .	142
4.4.1	Capteur de ligne ZigBee . . . . .	143
4.4.2	Passerelle IP/ZigBee . . . . .	144
4.4.3	Flotte de robots ZigBee . . . . .	144
4.4.4	Bilan sur les applications de démonstration . . . . .	145
	<b>Conclusion</b> . . . . .	<b>149</b>
	<b>Bibliographie générale</b> . . . . .	<b>153</b>
	<b>Glossaire</b> . . . . .	<b>159</b>
	<b>Table des figures</b> . . . . .	<b>162</b>
	<b>Liste des tableaux</b> . . . . .	<b>167</b>



# Introduction

De nos jours, les réseaux de télécommunications sont présents dans de très nombreux domaines. Que ce soit dans la production, l'espace ou la domotique, les réseaux sont devenus incontournables. Le multiplexage des données sur un bus unique permet de limiter considérablement les câbles, abaissant globalement les coûts, tant à l'installation qu'à la maintenance. Les câbles sont généralement inesthétiques, coûteux, difficiles à installer et peu pratiques à l'usage. De plus, dans des domaines industriels où l'appareil est soumis à des contraintes vibratoires très importantes, leur connectique représente une faiblesse qui augmente considérablement le risque de panne. Enfin, dans certains domaines comme l'aéronautique ou le spatial, les câbles représentent un poids non négligeable qu'il est absolument nécessaire de limiter par une infrastructure plus légère. Dans cette démarche de forte limitation des câbles, le *sans fil* permet d'aller encore plus loin que le bus de données. Le support de transmission sans fil étant immatériel, il permet de gagner largement en poids, il peut traverser des cloisons sans nécessité de perçage et permet une forte mobilité des appareils connectés. En revanche, si les gains sont importants sur ces aspects, l'utilisation du *sans fil* amène son lot de contraintes qui empêche la transposition directe des mécanismes utilisés sur les bus filaires au *sans fil*. Par exemple, le phénomène d'*aveuglement* empêche la détection des *collisions* en temps réel, la forte variabilité des caractéristiques du médium radio entraîne une grande incertitude sur la fiabilité des liens et la forte atténuation du médium immatériel implique des portées limitées qui ne sont pas toujours simples à appréhender, même avec beaucoup d'expérience. Les perspectives de recherches, tant sur la couche *physique* que sur les protocoles de niveaux supérieurs, restent toujours importantes dans ce domaine.

Pourtant, dans le cadre d'une application communicante de type industriel comme un réseau de surveillance multi-capteurs, la transmission de l'information doit être fiable. Mieux que des performances moyennes globalement correctes, certaines applications peuvent nécessiter un moyen d'acheminement de l'information qui soit en mesure de proposer des garanties, par exemple, sur le volume d'informations à transmettre dans un temps donné (débit) ou le délai de remise d'un message (latence). Il est nécessaire d'introduire des fonctionnalités de *Qualité de Service* pour la gestion du réseau. Pour une application temps réel utilisant des mesures issues de capteurs, l'aspect temporel est primordial car la donnée représentant la grandeur physique mesurée perd sa pertinence avec le temps ; si le résultat de la mesure arrive avec un délai trop important, il doit être ignoré par l'applicatif car obsolète. Il est donc nécessaire, pour cette catégorie d'applications à très fortes contraintes temporelles, de concevoir des méthodes de transmission permettant de garantir le délai du transport de l'information. Bien qu'utilisant un médium imparfait, il est cependant possible de fiabiliser une transmission sans fil. Cette fiabilisation doit être un souci permanent pour le concepteur, à chaque niveau de la *pile protocolaire*. Au niveau *physique*, il existe des techniques reconnues et largement utilisées telles que le codage du canal ou la redondance des données émises. Cependant, même si ces techniques permettent de rendre plus robuste la transmission radio, elles ne permettent pas de garantir que 100% des transmissions seront reçues sans erreur. Il est alors parfois nécessaire de prévoir la retransmission des données erronées, ce qui, dans le cadre d'une application à fortes contraintes temporelles, entraîne des délais aléatoires qui peuvent se révéler gênants et qui accentueront l'aspect aléatoire de la couche physique.

Pour les couches supérieures, le premier élément susceptible d'introduire des incertitudes « au dessus » de la couche physique est l'algorithme de la *méthode d'accès au médium*. La méthode d'accès au médium (MAC) est le protocole mis en place pour gérer les accès concurrents au médium partagé ; il se situe directement au-dessus de la couche physique et permet de minimiser, voire de supprimer, le phé-

nomène des collisions, c'est-à-dire d'empêcher que deux émetteurs utilisent le même médium en même temps. En effet, si plusieurs émetteurs utilisent le médium partagé au même moment, il est probable que la résultante de la communication soit inaudible pour tous les destinataires. Selon les objectifs attendus, le protocole de la méthode d'accès peut être conçu de différentes manières, l'essentiel étant en premier lieu, que toutes les entités susceptibles de s'échanger des messages suivent le même protocole. Pour notre part, nous nous sommes attachés, dans ces travaux de thèse, aux méthodes d'accès présentant *un caractère fortement déterministe*, c'est-à-dire en mesure de proposer des bornes (minimum, maximum) sur l'aspect temporel de la remise des messages.

Au delà de l'aspect purement temporel, il nous a semblé intéressant d'implémenter une méthode d'accès au médium déterministe, dans un réseau composé d'unités contraintes énergétiquement. En effet, le caractère déterministe de la méthode d'accès permet de limiter les collisions, entraînant par la même occasion des économies sur le plan énergétique puisque le nombre de réémissions est considérablement limité. La prise en compte de l'aspect énergétique au niveau MAC permet d'entrevoir des perspectives de recherche novatrices : les techniques conventionnelles telles que l'*écoute permanente* d'un médium inutilisé, le *bourrage* ou les *retransmissions* constituent des pertes d'énergie évidentes qu'il est possible de réduire par l'optimisation des protocoles, notamment par l'introduction d'une méthode d'accès déterministe au médium.

Si l'on considère le réseau dans son ensemble, c'est-à-dire au-delà du lien de transmission direct entre deux émetteurs-récepteurs, il est nécessaire, afin d'assurer une *Qualité de Service globale à tout le réseau*, d'implémenter une couche *réseau* proposant des fonctionnalités de QoS au dessus de la couche MAC. En effet, conformément à la description faite par le modèle OSI, si la couche MAC, au travers de la méthode d'accès, est chargée de la QoS sur le lien physique, la couche *réseau* doit l'assurer sur tout le réseau, globalement. Ces deux couches, du point de vue de la gestion de la QoS, sont intimement liées. La couche MAC doit permettre la mise en œuvre de fonctionnalités telles que la réservation du médium et la différenciation de service ; elle doit aussi « remonter » des statistiques sur l'utilisation du médium en temps réel à la couche réseau pour lui permettre d'affiner la pertinence du protocole de routage. En résumé, elle doit permettre à chaque nœud du réseau d'accéder au médium selon ses besoins tout en minimisant les accès inutiles puisque le médium radio est une ressource rare. Mieux, elle doit faire en sorte que chaque transmission initiée sur le médium et réussie (c'est-à-dire sans erreur de transmission) parvienne jusqu'au niveau de l'*application* : il est en effet dommageable qu'un message transmis avec succès au niveau *physique* soit ensuite détruit dans les couches supérieures. . .

Parmi les nombreuses technologies existantes (technologies de transmission sans fil et protocoles de gestion de Qualité de Service) qui seront présentées dans le premier chapitre de ce mémoire, trop peu proposent les fonctionnalités qui viennent d'être évoquées. La plupart des technologies de réseaux sans fil de type WLAN/WPAN implémentent des méthodes d'accès au médium fonctionnant en mode *Best-Effort*, basées sur des algorithmes aléatoires, empêchant ainsi la couche réseau de mettre en œuvre des possibilités fines de gestion de QoS et liées à la technique d'accès au médium. Certaines méthodes d'accès proposent néanmoins des fonctionnalités de *différenciation de service* et de *réservation de l'accès au médium* mais ces méthodes ne permettent pas d'en garantir totalement l'accès. De plus, bien que décrites dans les normes et les spécifications, ces fonctionnalités ne sont généralement pas implémentées sur les produits commercialisés (802.11 PCF, 802.15.1), empêchant toute étude de performances réelles par prototypage.

A ce titre, la technologie IEEE 802.15.4/ZigBee, qui a été utilisée et améliorée dans le cadre des travaux de la thèse, présente des caractéristiques intéressantes de notre point de vue, notamment au niveau de la gestion de l'énergie et de la couche MAC. En effet, cette technologie WPAN est optimisée pour les « petits transferts de données » et correspond tout à fait aux attentes, sous l'angle énergétique, d'une application robotique de réseau de capteurs. De plus, la disponibilité matérielle de modules totalement reprogrammables, y compris au niveau de la couche MAC, nous a permis d'entrevoir des possibilités avantageuses de prototypage réel.

Le mémoire est structuré de la manière suivante : dans le premier chapitre, nous présentons un état de l'art sur les technologies de réseaux sans fil et sur les méthodes existantes pour gérer la Qualité de Service. Plusieurs technologies, normes et protocoles, sont présentés et comparés ; cette présentation n'est pas exhaustive, le but étant ici de dresser un état de l'art général de l'existant. A la fin de ce premier chapitre, nous présentons les principales techniques de méthodes d'accès au médium existantes, avec ou sans possibilités de QoS.

Dans le second chapitre, nous présentons les normes et spécifications IEEE 802.15.4 et ZigBee sur lesquelles sont basés les travaux élaborés pendant la thèse. L'aspect accès au médium proposé pour cette technologie sera particulièrement détaillé et des failles identifiées comme critiques, au regard de nos exigences quant au déterminisme de la méthode d'accès, seront discutées en fin de chapitre.

Dans le troisième chapitre, nous proposons une nouvelle méthode d'accès entièrement déterministe, c'est-à-dire proposant des garanties sur l'accessibilité du médium régulière et prédéterminée par un contrat. La méthode d'accès est dite « entièrement déterministe » car même la phase d'association au réseau peut être réalisée sur ce principe. Bien que cette méthode d'accès puisse être généralisée à tout type de WLAN/WPAN, elle s'appuie cependant sur les failles de IEEE 802.15.4, identifiées comme telles dans le chapitre 2.

Dans le quatrième chapitre, nous présentons les résultats que nous avons obtenus concernant la méthode d'accès déterministe présentée dans le chapitre 3. Pour tester, valider et évaluer les performances de notre contribution, nous avons réalisé plusieurs études : modélisation par Réseaux de Petri, simulation et prototypage réel. Les performances du réseau et notre analyse quant à l'efficacité de la méthode d'accès au médium seront détaillées dans ce quatrième et dernier chapitre.



# Chapitre 1

## Les réseaux sans fil et la Qualité de Service

*Dans ce premier chapitre, nous dressons un état de l'art des différentes techniques de transmission réseau, en nous focalisant particulièrement sur les technologies sans fil et les méthodes liées à la Qualité de Service (QoS).*

*Dans un premier temps, nous présentons les principales technologies de réseaux sans fil en nous attachant à leurs caractéristiques majeures. Dans un deuxième temps, nous présentons une étude bibliographique sur les techniques de communication et les protocoles à Qualité de Service. Ces techniques étant malheureusement le plus souvent gérées au niveau Réseau (niveau 3 OSI), nous reviendrons dans un troisième et dernier temps sur les technologies de transmission sans fil et aborderons la problématique de la méthode d'accès (niveau 2 OSI); nous verrons alors comment une Qualité de Service peut être avantageusement mise en œuvre à ce niveau.*

---

<b>1</b>	<b>État de l'art des technologies de réseaux sans fil . . . . .</b>	<b>13</b>
1.1	Classification des technologies réseaux . . . . .	13
1.2	Les réseaux sans fil étendus (WWAN) . . . . .	14
1.3	Les réseaux sans fil métropolitains (WMAN) . . . . .	14
1.3.1	La Boucle Locale Radio (BLR) . . . . .	14
1.3.2	ETSI HiperMAN / IEEE 802.16, WiMAX . . . . .	15
1.3.3	IEEE 802.20, Mobile Broadband Wireless Access (MWBA) . . . . .	16
1.3.4	Le phénomène des réseaux sans fil ruraux et des réseaux sans fil coopérants . . . . .	16
1.4	Les réseaux locaux sans fil (WLAN) . . . . .	16
1.4.1	ETSI HiperLAN . . . . .	17
1.4.1.1	HiperLAN/1 (High PERformance LAN, première version) . . . . .	17
1.4.1.2	HiperLAN/2 (High PERformance LAN, seconde version) . . . . .	17
1.4.2	IEEE 802.11, WiFi . . . . .	18
1.4.2.1	La norme initiale . . . . .	18
1.4.2.2	Topologies . . . . .	18
1.4.2.3	Les évolutions de 802.11 . . . . .	19
1.5	Les réseaux personnels sans fil (WPAN) . . . . .	21
1.5.1	IrDA . . . . .	21
1.5.2	HomeRF . . . . .	22
1.5.3	IEEE 802.15.1, Bluetooth . . . . .	22
1.5.4	IEEE 802.15.3, Ultra Wide Band (UWB) . . . . .	24
1.5.5	IEEE 802.15.4, ZigBee . . . . .	25

1.6	Les réseaux de capteurs (WSN) . . . . .	25
<b>2</b>	<b>La Qualité de Service dans les réseaux . . . . .</b>	<b>26</b>
2.1	Présentation générale . . . . .	26
2.1.1	Qu'est ce que la Qualité de Service? . . . . .	26
2.1.2	QoS vs. Surdimensionnement . . . . .	26
2.2	Notions fondamentales et principes de QoS . . . . .	27
2.2.1	La prédictabilité du trafic . . . . .	27
2.2.2	Le contrôle d'admission . . . . .	28
2.2.3	Les degrés de Qualité de Service . . . . .	28
2.3	Quelques exemples de protocoles à QoS . . . . .	29
2.3.1	Asynchronous Transfer Mode (ATM) . . . . .	29
2.3.2	IntServ . . . . .	30
2.3.2.1	Profils de QoS au niveau utilisateur . . . . .	30
2.3.2.2	Le protocole de signalisation RSVP . . . . .	30
2.3.2.3	Conclusion : bénéfices et difficultés liés à IntServ . . . . .	31
2.3.3	DiffServ . . . . .	32
2.4	Conclusion . . . . .	32
<b>3</b>	<b>Problématique de l'accès au médium partagé . . . . .</b>	<b>33</b>
3.1	Accéder au médium partagé : les principes de base . . . . .	33
3.2	Les principales techniques d'accès au médium . . . . .	34
3.2.1	CSMA/CA (IEEE 802.11 DCF) . . . . .	34
3.2.2	EY-NPMA (HiperLAN/1) . . . . .	37
3.3	Introduire de la QoS au niveau accès au médium . . . . .	39
3.3.1	TDMA Dynamique (HiperLAN/2) . . . . .	39
3.3.2	Point Coordination Function (IEEE 802.11 PCF) . . . . .	40
3.3.3	Enhanced DCF et Hybrid CF (IEEE 802.11e) . . . . .	41
3.3.3.1	Enhanced DCF . . . . .	42
3.3.3.2	Hybrid CF . . . . .	42
3.3.3.3	Autres améliorations . . . . .	43
3.3.4	Bluetooth . . . . .	43
<b>4</b>	<b>Conclusion . . . . .</b>	<b>44</b>

---

# 1 État de l'art des technologies de réseaux sans fil

Dans cette partie, nous présentons les différentes technologies de réseaux sans fil existantes. Les caractéristiques principales sont évoquées : topologies, fréquences, débit bande de base, portées théoriques, modulations et étalement de spectre.

## 1.1 Classification des technologies réseaux

Les technologies réseaux sont fréquemment classées en quatre catégories. Cette classification tient essentiellement compte de la portée des dites technologies :

- les réseaux personnels (PAN<sup>◇</sup>) : ils concernent l'entourage immédiat d'une personne (quelques mètres),
- les réseaux locaux (LAN<sup>◇</sup>) : ils concernent un environnement de vie plus étendu que les réseaux personnels comme une maison, une entreprise ou un campus (quelques dizaines de mètres à quelques kilomètres),
- les réseaux métropolitains (MAN<sup>◇</sup>) : ils visent à couvrir une région étendue comme une ville (plusieurs kilomètres),
- les réseaux étendus (WAN<sup>◇</sup>) : ils visent à couvrir une zone très vaste comme un pays, une région du globe ou toute la planète.

Les réseaux sans fil (en anglais *Wireless Networks*) n'échappent pas à cette classification : on parle souvent de WPAN<sup>◇</sup> (par exemple la technologie *Bluetooth*), WLAN<sup>◇</sup> (par exemple *WiFi*), WMAN<sup>◇</sup> (par exemple *WiMAX*) ou WWAN<sup>◇</sup> (par exemple les réseaux satellites), comme l'illustre la figure 1.1.

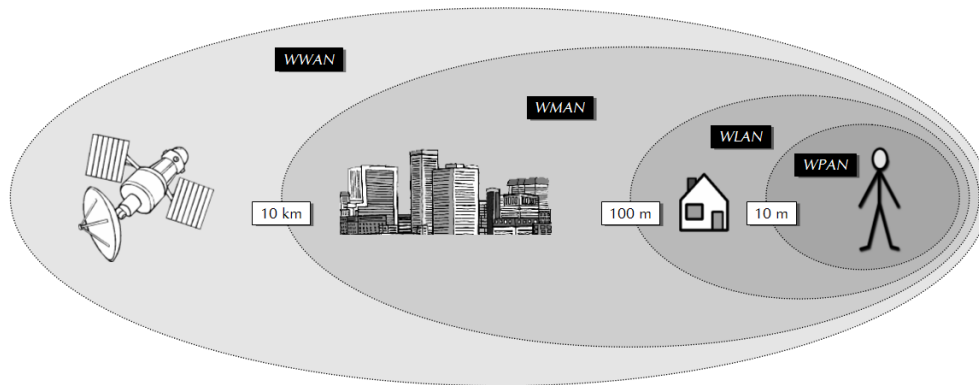


FIG. 1.1 – Différents types de réseaux sans fil et portées typiques

Bien que la portée constitue le critère de classement le plus répandu, il peut être intéressant de classer les technologies réseaux selon d'autres paramètres comme *le débit proposé*, *l'énergie nécessaire au fonctionnement*, *le coût de mise en œuvre et d'entretien*, *la simplicité d'utilisation* ou bien encore *la fiabilité*. Des compromis différents sont faits entre ces caractéristiques ; ils donnent lieu à une multitude de techniques – certaines complémentaires, d'autres concurrentes. Ce grand nombre de produits disponibles permet dès lors de proposer une (voire plusieurs) solution(s) à une problématique donnée. Concernant plus particulièrement les technologies de réseaux sans fil, et de par le caractère embarqué des équipements visés, un autre classement, fondé sur la consommation énergétique, serait aussi très pertinent.

Nous allons à présent proposer un état de l'art non exhaustif des principales technologies de communication réseaux sans fil que l'on peut rencontrer sur le marché.

## 1.2 Les réseaux sans fil étendus (WWAN)

Cette catégorie compte assez peu de technologies à l'heure actuelle ; en effet, la plupart des réseaux étendus sont des réseaux filaires<sup>1</sup>. Cette constatation s'explique très simplement par la physique des transmissions : lorsque l'on souhaite transporter des informations sur un médium immatériel, il est généralement admis de faire porter ces informations par une *fréquence porteuse*. Plus le débit des informations est élevé, plus la fréquence porteuse doit l'être également. Or, on constate qu'à partir d'une fréquence porteuse de 50 MHz, les ondes électromagnétiques ne se réfléchissent plus sur les différentes couches de l'atmosphère. Pour ces réseaux haut débits sans fil, il est donc impossible d'atteindre des portées autres que « à vue ».

Les seules technologies de WWAN disponibles sont des technologies utilisant les satellites *géostationnaires* ou en *orbite basse* pour relayer l'information entre plusieurs points du globe. Il existe plusieurs standards de réseaux utilisant les satellites [PUJO 95] comme VSAT<sup>◇</sup> [HOWE 87] (utilisés pour la transmission de transactions bancaires) ou DVB-S [SMER 04] (norme grand public pour les diffusions de télévision, radio et service de données par satellite). Nous ne détaillerons pas plus longuement ces techniques qui dépassent le champ de la thématique étudiée.

## 1.3 Les réseaux sans fil métropolitains (WMAN)

Les réseaux métropolitains ont pour objectif de créer un ensemble de liens de communication sur une zone étendue de la taille d'une ville ou d'une région. Ces liens peuvent servir à interconnecter plusieurs sites d'une même entreprise ou d'une administration. Ils peuvent être l'initiative d'une collectivité locale : certaines villes en mal d'investisseurs et de créateurs d'entreprises participent pour une part importante au financement de réseaux très haut débit ; c'est par exemple le cas de la ville de Pau, dans les Hautes Pyrénées Françaises, avec le projet *Pau Broadband Country* [CAPP 02], un réseau optique FTTH<sup>◇</sup> opérationnel depuis 2003 qui couvre la ville et ses environs, donnant un accès à 100 Mbits/s pour ses abonnés. Bien que certains réseaux dits métropolitains soient privés et ne constituent qu'un réseau local étendu, la plupart d'entre eux sont en fait des *réseaux d'accès* (ils permettent d'accéder à un autre réseau).

En ce qui concerne les réseaux *sans fil métropolitains*, ils sont généralement utilisés dans les campagnes pour palier à la faiblesse des accès de type xDSL<sup>◇</sup>, qui sont plutôt destinés aux zones à forte densité de population. A la fin des années 90, un déploiement de WMAN avait été entamé en France avec le projet de Boucle Locale Radio, mais c'est sans nul doute le standard IEEE<sup>◇</sup> 802.16 [IEE1 04], plus connu sous le nom de *WiMAX*, qui devrait s'imposer dans les années à venir. Dans cette catégorie, d'autres standards ont aussi été développés, ils sont évoqués ici à titre informatif.

### 1.3.1 La Boucle Locale Radio (BLR)

La *Boucle Locale* est la partie d'un réseau de télécommunications située entre la prise téléphonique de l'abonné et le central téléphonique : c'est la partie terminale du réseau d'accès de l'opérateur que l'on désigne parfois par « *le dernier mile* ». On parle de *boucle* car, historiquement, le réseau téléphonique commuté fonctionnait grâce à une boucle de courant qui alimentait les postes téléphoniques des usagers.

Dans le cas de la *Boucle Locale Radio*, ou BLR<sup>◇</sup>, les données sont acheminées par les ondes hertziennes et non par la paire de cuivre. C'est une solution économique dans la mesure où il n'est pas nécessaire de « poser » des câbles et d'effectuer des travaux de type Génie Civil. La BLR permet donc de compléter la desserte filaire traditionnelle dans les zones où la densité de clients potentiels est faible.

Il est courant de désigner par BLR toute technologie répondant à l'objectif qui vient d'être énoncé ;

---

<sup>1</sup>Contrairement à certaines idées reçues, la majeure partie du trafic du réseau Internet, par exemple, est véhiculée par des réseaux filaires (en général optiques), même pour accéder à des serveurs situés à l'opposé de la planète.



cependant, la technologie qui était la plus répandue jusqu'au milieu des années 2000 était le LMDS<sup>◇</sup>. Le LMDS est une technologie d'accès radio large bande. Elle utilise des fréquences très élevées (situées entre 26 et 29 GHz pour la France<sup>2</sup>). Le LMDS est principalement utilisé pour créer des liens de type point à multipoints (une station de base communique avec plusieurs stations clientes) sur des distances pouvant atteindre 8 kilomètres et permettant de donner un accès réseau à 10 Mbits/s environ. Il est également possible de réaliser des liaisons point à point grâce à des antennes directionnelles ; dans ce cas, la portée peut être grandement améliorée et le débit ne profite qu'à une paire d'entité communicantes. On parlera alors de *faisceau hertzien*.

La technologie WiMAX, qui sera présentée ci-après, est aujourd'hui généralement préférée au LMDS car elle offre globalement de meilleures performances, notamment en terme de débit. De plus, WiMAX a été conçu pour proposer à l'utilisateur des fonctionnalités de mobilité, ce qui n'est pas le cas de LMDS.

### 1.3.2 ETSI HiperMAN / IEEE 802.16, WiMAX

WiMAX est le consortium qui est chargé de la promotion des technologies de réseaux sans fil métropolitains IEEE 802.16 [IEE1 04] et ETSI HiperMAN [ETS1 06] [ETS2 06] ; il prône la convergence de ces deux standards pour une technologie unique et interopérable. A ce titre, le consortium WiMAX a créé le *label* « WiMAX » qui certifie que l'équipement labellisé est approuvé selon les critères définis par le consortium et qu'il est compatible avec les autres équipements labellisés, quelque soit le fabricant du matériel<sup>3</sup>.

De part leurs concepteurs communs, les normes HiperMAN et 802.16 sont très semblables. HiperMAN est une norme proposée par le comité européen de standardisation des télécommunications (ETSI) [ETSI] qui a pour objectif de proposer une normalisation des réseaux sans fil métropolitains, essentiellement pour raccorder les PME et les particuliers à un réseau haut débit. HiperMAN propose d'utiliser la bande des 2 à 11 GHz et prévoit des mécanismes d'adaptation automatique de la puissance d'émission, de la modulation et du codage pour fiabiliser la transmission dans les conditions les plus difficiles, permettant ainsi d'envisager des liens de transmission entre deux stations qui ne sont pas à *vue* (en anglais *non-line-of-sight*) comme un terminal dans un bâtiment. Le standard propose également un multiplexage optimisé à base de TDD<sup>◇</sup> et de FDD<sup>◇</sup>. La technologie permet le transport natif de paquets IP<sup>◇</sup> ou de cellules ATM<sup>◇</sup> pour des applications multimédia (voix, vidéo, téléphonie) [GOLD 01] [HOYM 01].

De même qu'HiperMAN, IEEE 802.16 propose lui aussi un standard pour les réseaux sans fil métropolitains haut débits. Comme tous les autres standards issus du comité de standardisation 802 [IEEE] de l'organisme de standardisation américain IEEE, ce standard est le résultat d'un travail mené par le groupe de discussion (*Tasking Group*) désigné par « TG16 ». Il spécifie les couches *physique* et *liaison de données*, c'est à dire les deux premiers niveaux du modèle OSI (*Open System Interconnection*, proposé par l'ISO<sup>◇</sup>). 802.16 propose un débit bande de base de 70 Mbits/s sur un rayon d'une dizaine de kilomètres, la portée pouvant bien entendu varier avec les éventuels obstacles. Plusieurs bandes de fréquences sont proposées (2,5 à 3,5 GHz, 10 à 66 GHz) ainsi que plusieurs modulations basées sur le principe de l'OFDM<sup>◇</sup> qui consiste à répartir plusieurs flux de données sur autant de sous-porteuses. Comme les données sont émises en bande étroite, elles sont moins sujettes à la variabilité des caractéristiques du médium radio, en particulier pour des transmissions présentant des échos importants (multi-trajets).

Contrairement à la BLR, WiMAX propose d'introduire des fonctionnalités de *mobilité* dans son réseau : un terminal WiMAX peut se déplacer tout en conservant un accès fiable au réseau. Cette fonctionnalité est introduite par la norme IEEE 802.16e (*Mobile WiMAX*) qui a été ratifiée le 7 décembre 2005. L'innovation est de taille car cette fonctionnalité permettrait à la technologie WiMAX d'entrer en concurrence directe avec les technologies de réseaux de téléphonie mobile comme GPRS<sup>◇</sup>, EDGE<sup>◇</sup> ou UMTS<sup>◇</sup>. Par extension, on assisterait là à une technologie issue du monde Internet qui viendrait

<sup>2</sup>Comme pour toute technologie utilisant les ondes électromagnétiques, les fréquences peuvent varier selon la région du globe où elles sont utilisées. Aux États-Unis, LMDS utilise des fréquences situées entre 31,0 et 31,3 GHz.

<sup>3</sup>Nous verrons plus loin que le label « WiMAX » n'est pas le seul label dans le monde des réseaux ; « Bluetooth » et « Wi-Fi » sont deux labels qui concernent respectivement les normes IEEE 802.15.1 et IEEE 802.11.

marcher sur les « plates bandes » du monde des opérateurs de télécommunications ; il y a là un défi économique considérable. Notons cependant que comme la plupart des technologies WMAN, l'utilisation de WiMAX n'est généralement pas libre. WiMAX peut être utilisé seulement sous couvert d'une licence délivrée par l'autorité de régulation des télécommunications de la région où il est utilisé, au même titre qu'un réseau d'opérateur. En France, cette licence est attribuée par l'ARCEP<sup>◇</sup> [ARCE], anciennement ART<sup>◇</sup>. Notons que concernant la France, l'ARCEP n'a pas autorisé pour l'instant l'utilisation de *Mobile WiMAX* (802.16e).

### 1.3.3 IEEE 802.20, Mobile Broadband Wireless Access (MWBA)

Le comité de standardisation 802 (LAN/WAN) de l'IEEE compte un groupe de discussion sur les réseaux d'accès sans fil large bande et mobiles (*Mobile Broadband Wireless Access*, MWBA), le TG20. L'objectif de ce groupe de discussion est l'élaboration d'un standard pour une couche physique et une couche liaison de données, pour un réseau mobile sans fil haut débit *orienté paquets*, en vue d'un interfaçage direct avec IP. Les objectifs du TG20 sont assez semblables de ceux du TG16e (*Mobile WiMAX*).

Les principales caractéristiques du réseau MWBA sont les suivantes :

- mobilité des nœuds jusqu'à 250 km/h,
- débit 1 Mbits/s (au niveau IP),
- couche physique utilisant la bande des 3,5 GHz, soumise à licence,
- mode paquets, itinérance de niveau IP (*IP roaming & handoff*).

A ce jour, les caractéristiques évoquées restent un objectif à atteindre dans la mesure où aucune implémentation de MWBA n'a encore été réalisée. IEEE 802.20 est encore à l'état de *draft* (brouillon) ; il a été adopté le 18 janvier 2006. Cependant, les travaux du TG20 auraient été suspendus par l'IEEE en juin 2006, ceci pour répondre aux accusations portées à son encontre selon lesquelles la société Qualcomm aurait pris une part trop importante dans les travaux de 802.20. Dans tous les cas, cette norme promet quelques ambitieuses avancées dans le domaine des réseaux mobiles haut débit ; ces dernières demandent cependant à être vérifiées expérimentalement.

### 1.3.4 Le phénomène des réseaux sans fil ruraux et des réseaux sans fil coopérants

Il peut être intéressant d'évoquer un phénomène nouveau, d'origine plus économique et sociale que technique : le phénomène des réseaux sans fil ruraux et des réseaux sans fil coopérants.

On assiste depuis quelques années à la création de réseaux de communication, indépendants des opérateurs [FON], parfois même communautaires [TSF], qui ont pour but de partager des ressources communes ou tout simplement, de recréer un réseau dans des régions où l'accès au réseau Internet n'est pas disponible. C'est notamment le cas dans les régions rurales des pays occidentaux [RAN] ou dans les zones métropolitaines des pays en voie de développement.

Techniquement, ces réseaux proposent un accès de quelques Mbits/s sur une zone étendue, mais en se basant sur des technologies très bon marché comme 802.11 qui sera évoqué ci-après. Bien que ce type de réseaux n'offre aucune garantie, notamment en terme de fiabilité, ils proposent cependant une alternative que certains opérateurs sont parfois amenés à reproduire ou racheter.

## 1.4 Les réseaux locaux sans fil (WLAN)

Les réseaux locaux sont de taille plus réduite que les réseaux métropolitains. Ils visent généralement la couverture d'une zone de quelques centaines de mètres maximum comme une entreprise ou un petit campus. La majeure partie des réseaux dits « locaux » servent à partager des ressources communes

comme les documents ou les imprimantes. Ils constituent également un accès à des réseaux d'un autre type comme Internet. De plus, l'explosion récente des applications de type « Voix sur IP » (VoIP) apporte une nouvelle utilité aux réseaux locaux qui acheminent désormais les communications téléphoniques jusqu'au poste compatible VoIP de l'abonné. Par exemple, de grandes entreprises comme Airbus [CHIC 05] passent désormais des contrats avec des opérateurs de téléphonie sur IP pour une réorganisation complète de leur réseau de téléphonie interne. Nous assistons là à une convergence des divers réseaux de télécommunications vers un réseau unique basé sur IP.

Les technologies de réseaux locaux sans fil se développent elles aussi, parallèlement aux technologies filaires. Alors que le *sans fil* permet, dans le cadre des réseaux métropolitains, l'acheminement du réseau dans les zones difficiles (les WMAN constituent une alternative économique à un problème social), le réseau local sans fil va plutôt répondre à un objectif de confort pour l'utilisateur en lui permettant d'être plus autonome et mobile dans un environnement où le câblage peut rester néanmoins envisageable.

Dans la suite de cette partie, nous allons décrire les principales technologies WLAN et exposer leurs caractéristiques essentielles.

### 1.4.1 ETSI HiperLAN

HiperLAN est une norme de réseau local sans fil mise au point par le comité RES10 (*Radio Equipment and Systems*) de l'ETSI [ETSI]. Quatre déclinaisons d'HiperLAN ont été proposées par le comité : HiperLAN/1, HiperLAN/2 [ETSI 00], HiperACCESS et HiperLINK. La norme HiperLAN décrit simplement l'interface radio, la couche physique et la couche liaison ; les spécifications de la norme HiperLAN concernent donc les deux premières couches du modèle OSI.

#### 1.4.1.1 HiperLAN/1 (High PERFORMANCE LAN, première version)

Le premier projet HiperLAN (HiperLAN/1) a été lancé en 1991, peu après les premiers travaux sur 802.11 et a été ratifié durant l'été 1996. HiperLAN/1 est une norme très complète qui propose une couche physique, une méthode d'accès et un protocole de routage<sup>4</sup> (*intra-forwarding*, basé sur un algorithme qui s'appellera plus tard OLSR<sup>◇</sup> [BADI 04]) pour élaborer des réseaux ad hoc maillés et spontanés. Cette technologie opère sur la bande des 5,1 GHz - 5,3 GHz et propose un débit bande de base de 23,5 Mbits/s, sur une portée de 50 mètres environ. Les modulations utilisées sont de type FSK<sup>◇</sup> et GMSK<sup>◇</sup>.

En 1996, les concepteurs de cette norme étaient certainement trop en avance sur leur temps et leur proposition d'avant-garde n'a pas reçu de soutien économique. Elle a rapidement été abandonnée au profit d'HiperLAN/2.

#### 1.4.1.2 HiperLAN/2 (High PERFORMANCE LAN, seconde version)

HiperLAN/2 [ETSI 00] a été ratifiée en février 2000. Cette seconde version est radicalement différente de la première : mode infrastructure, transport natif d'ATM et UMTS<sup>◇</sup>, plusieurs couches physiques pour s'adapter à l'environnement électromagnétique.

HiperLAN/2 a été la première technologie à mettre en œuvre la technique d'étalement de spectre OFDM. Elle propose plusieurs modulations (BPSK<sup>◇</sup>, QPSK<sup>◇</sup>, 16QAM<sup>◇</sup> et 64QAM<sup>◇</sup>) pour plusieurs débits (6, 9, 12, 18, 27, 36 et 54 Mbits/s) sur la bande des 5,4 GHz - 5,7 GHz. Une utilisation de la bande des 17 GHz aurait également été prévue. La couche physique d'HiperLAN/2 a ensuite été reprise par 802.11a, puis 802.11g, ce qui explique les similarités des fréquences et des débits en bande de base entre HiperLAN/2 et 802.11a/g.

<sup>4</sup>On parlera plutôt de relais de trames, puisque ce processus se situe au niveau 2 OSI.

HiperLAN/2 est intimement liée à ATM qui était prévu pour être le protocole de niveau réseau/transport au dessus d'HiperLAN/2. Outre le transport des cellules ATM, HiperLAN/2 peut aussi véhiculer nativement des données vidéo MPEG, des paquets IP et Firewire IEEE 1394, ainsi que la voix numérisée des téléphones cellulaires UMTS. HiperLAN/2 prévoit également une gestion de la mobilité pour des vitesses inférieures à 10 m/s par *handover* (changement de cellule) et propose une gestion de Qualité de Service par des communications établies en mode connecté. HiperLAN/2 a bénéficié en France du soutien de l'ART.

A l'opposé d'HiperLAN/1 qui est basée sur une topologie de réseau ad hoc, HiperLAN/2 propose un réseau sans fil d'infrastructure. Les informations échangées passent par un ou plusieurs points d'accès (AP, *Access Points*) qui gèrent l'accès au médium selon le principe du TDMA<sup>◇</sup> dynamique (repris ensuite dans IEEE 802.16) avec des transports d'informations en mode connecté.

Malheureusement pour lui, le RES10 n'a pas reçu autant de soutien que le groupe IEEE 802.11, son concurrent direct. La commercialisation d'équipements au standard HiperLAN/2 a été très limitée et le projet est abandonné aujourd'hui.

### 1.4.2 IEEE 802.11, Wi-Fi

IEEE 802.11 [IEE1 99] est un standard de réseau sans fil local proposé par l'organisme de standardisation Américain IEEE. La technologie 802.11 est généralement considérée comme la version sans fil de 802.3 (Ethernet). La norme originale avait été ratifiée en 1997 ; elle a depuis été largement amendée par de nombreux groupes de travail et même si les principes de base restent les mêmes depuis 10 ans – mode non connecté, CSMA/CA<sup>◇</sup>, 802.2 LLC<sup>◇</sup> – IEEE 802.11 est aujourd'hui la technologie de réseaux sans fil locaux la plus utilisée à travers le monde, sans doute du fait de sa grande simplicité, de son faible coût de mise en œuvre mais aussi grâce aux importants soutiens techniques et financiers dont elle a bénéficié.

#### 1.4.2.1 La norme initiale

Dans sa version initiale de 1997, 802.11 proposait trois couches physiques [MUHL 02] :

- Radio, à étalement de spectre par utilisation de séquences directes (DSSS<sup>◇</sup>), débit bande de base 1 Mbits/s et 2 Mbits/s,
- Radio, à étalement de spectre par utilisation de sauts de fréquences (FHSS<sup>◇</sup>), à 1,6 Mbits/s,
- Infrarouge, 1 ou 2 Mbits/s.

A l'heure actuelle, seule la couche physique radio à étalement de spectre par séquences directes (DSSS) perdure. Les deux autres (Radio FHSS et infrarouge), bien qu'ayant été implémentées et commercialisées à la fin des années 90, n'ont pas survécu. Depuis, les techniques ont été fortement améliorées et les dernières couches physiques basées sur le principe du MIMO<sup>◇</sup> permettent d'obtenir des débits de plusieurs centaines de Mbits/s.

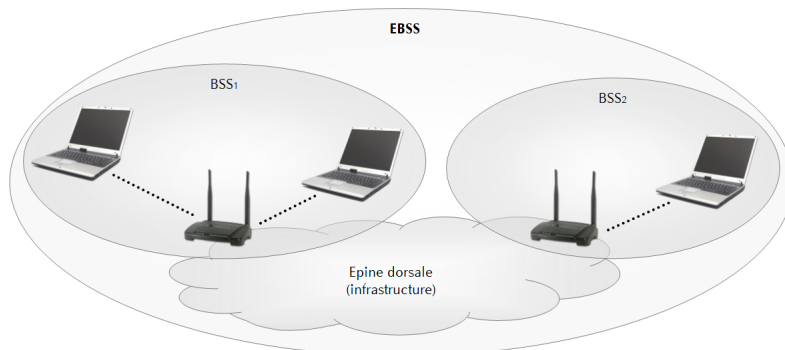
#### 1.4.2.2 Topologies

Dans le cadre de 802.11, deux topologies sont possibles aujourd'hui<sup>5</sup> :

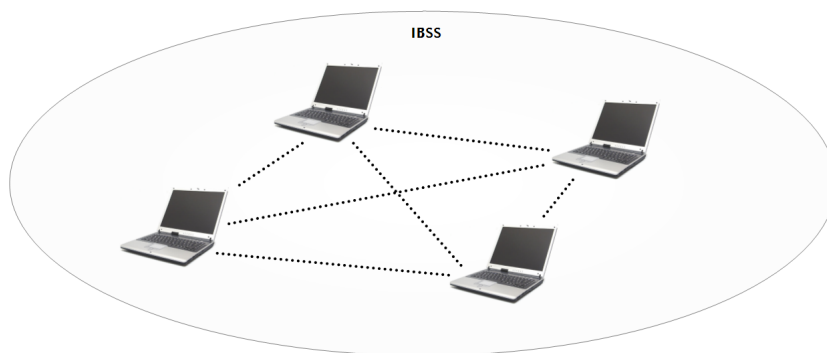
---

<sup>5</sup>Nous verrons ci-après qu'un groupe de travail de l'IEEE travaille actuellement sur une topologie ad hoc maillée de 802.11, un peu à la manière de HiperLAN/1.

- une topologie centralisée autour d'un ou plusieurs points d'accès, on parle alors de *mode infrastructure* (s'ils sont plusieurs, les points d'accès seront reliés par une épine dorsale de type Ethernet). Le réseau est alors formé de plusieurs BSS (*Basic Service Set*) qui forment ensemble un unique EBSS (*Extended Basic Service Set*). La figure 1.2 représente cette topologie,

FIG. 1.2 – Topologie *Infrastructure* de 802.11

- une topologie sans élément central, sans hiérarchie, où chaque station a le même rôle ; on parle dans ce cas de *mode ad hoc*. Le réseau formé est appelé IBSS (*Independent Basic Service Set*). La figure 1.3 représente cette topologie.

FIG. 1.3 – Topologie *Ad-hoc* de 802.11

#### 1.4.2.3 Les évolutions de 802.11

Nous l'avons vu plus haut, le standard initial 802.11 a été amendé par de nombreux groupes de travail. Chaque groupe de travail fait des propositions qui donnent lieu à des améliorations de la norme. Ils sont cités ici :

- **802.11a** propose 8 canaux dans la bande des 5 GHz. Cette proposition permet d'atteindre un débit bande de base de 54 Mbits/s sur une portée d'une vingtaine de mètres environ par utilisation du principe de l'OFDM (déjà évoqué dans le paragraphe 1.3.2 sur HiperMAN et WiMAX).
- **802.11b** propose une amélioration de la norme initiale en introduisant la modulation CCK<sup>◇</sup> dans la bande des 2,4 GHz. Deux nouveaux débits sont alors disponibles : 5,5 Mbits/s et 11 Mbits/s sur une portée de quelques dizaines de mètres environ. Ratifiée en septembre 1999, 802.11b est l'amendement de 802.11 qui a donné sa popularité au WiFi. Bien que 802.11b soit encore largement utilisé, il est maintenant supplanté par 802.11g.

- **802.11c** propose une modification de la norme 802.1d existante pour les réseaux filaires afin de la transposer à 802.11. Elle permet une normalisation de l'interconnexion de niveau 2 (pont) entre un réseau filaire et un réseau Wi-Fi.
- **802.11d** propose un protocole d'échange d'informations sur les fréquences et les puissances d'émission en vue d'une utilisation dans chaque région du monde, quelque soit le pays d'origine du matériel.
- **802.11e** propose des outils de Qualité de Service. Les travaux spécifiques de ce groupe de travail seront détaillés plus loin dans ce chapitre 1.
- **802.11f** est une recommandation qui propose une extension pour la communication entre points d'accès compatibles 802.11 par le protocole IAPP (*Inter-Access Point Protocol*) en introduisant des capacités de changement de cellules et d'équilibrage des charges (*load-balancing*).
- **802.11g** constitue une amélioration directe de 802.11b en proposant un débit bande de base de 54 Mbits/s sur la bande des 2,4 GHz. Ce gain en débit est réalisé en reprenant le concept de l'étalement de spectre par OFDM utilisé dans 802.11a. Toutefois, 802.11g garde une compatibilité avec 802.11b, ce qui signifie que des matériels conformes à la norme 802.11g peuvent fonctionner en 802.11b.
- **802.11h** propose des améliorations pour pallier au futur problème de la sur-utilisation des fréquences dédiées à 802.11. Ce groupe de travail propose d'une part une possibilité de sélection dynamique de fréquence appelée *DFS* pour *Dynamic Frequency Selection*, qui permet de choisir le canal le moins perturbé, et d'autre part le contrôle de puissance *TPC* pour *Transmit Power Control*, qui permet à l'émetteur de réduire sa puissance d'émission au minimum nécessaire.
- **802.11i** est la norme qui spécifie la solution de chiffrement désignée communément par le label *WPA2*<sup>◇</sup>. Elle s'appuie sur TKIP<sup>◇</sup> et AES<sup>◇</sup> et, contrairement à WPA, permet aussi bien la sécurisation des réseaux en mode infrastructure comme en mode ad hoc. Deux déclinaisons sont possibles : le *WPA personal* qui repose sur la connaissance d'une clé commune (comme le WEP<sup>◇</sup>, mais qui peut être de longueur variable, contrairement à ce dernier) et le *WPA enterprise* qui repose sur 802.1x avec un serveur d'authentification, par exemple de type RADIUS.
- **802.11IR** normalise une couche physique de type infrarouge pour 802.11. Elle utilise des signaux optiques de la bande des 800-900 nm et propose deux modulations de type *PPM* (*Pulse Position Modulation*) : 16-PPM qui permet d'atteindre un débit de 1 Mbits/s et 4-PPM pour 2 Mbits/s. Au départ soutenue pour ses vertus de sécurité – les signaux infrarouges ne traversent pas les cloisons – elle a rapidement été abandonnée au profit de 802.11a et 802.11b qui présentent tous deux des débits plus importants.
- **802.11j** propose une couche physique spécifique pour satisfaire à la réglementation japonaise. Très proche de 802.11a (OFDM, 54 Mbits/s, etc.), elle travaille dans la bande 4,9 GHz – 5 GHz.
- **802.11n** propose un débit bande de base de 540 Mbits/s sur une portée de 50 mètres environ grâce à l'utilisation conjointe des techniques MIMO et OFDM. Elle propose l'utilisation des deux bandes de fréquences 2,4 GHz (comme 802.11b et 802.11g) et 5 GHz (comme 802.11a). Comme 802.11g, cette norme reste compatible avec 802.11 ; de plus, elle reprend les concepts de 802.11e pour la gestion de la Qualité de Service, de 802.11i pour la sécurité et de 802.11f pour la gestion des *handovers*. A l'heure où ces lignes sont écrites, 802.11n est encore à l'état de *draft* (brouillon), suite au résultat inattendu du vote de l'IEEE le 2 Mai 2006. Elle devrait être ratifiée courant 2007. Cependant, il est clair que cette norme constituera le prochain standard en terme de WLAN.
- **802.11s** propose une extension à la méthode d'accès au médium de 802.11 ainsi qu'une méthode de routage pour les réseaux ad hoc de type *Mesh* (réseaux ad hoc maillés). La diffusion des routes est assurée par une méthode d'inondation et un algorithme très proche d'OLSR. Comme 802.11n,

la norme 802.11s est actuellement en cours d'élaboration et n'a pas encore été ratifiée. Elle devrait l'être dans le courant de l'année 2008<sup>6</sup>.

- **802.11r** propose d'introduire des fonctionnalités supplémentaires pour permettre le saut de cellule rapide d'un BSS à l'autre. Il devrait permettre aux terminaux mobiles de passer d'une cellule à une autre sans perte d'accès au réseau.

## 1.5 Les réseaux personnels sans fil (WPAN)

Les réseaux personnels concernent les liaisons de données en rapport direct avec l'individu et son entourage immédiat (télécommande, appareillage électroménager et accessoires, etc.). Comme pour les WLAN, le confort de l'utilisateur (mobilité, absence d'infrastructure lourde, etc.) est souvent la motivation première des concepteurs des produits utilisant ces technologies. Cependant, on note une certaine aptitude de quelques WPAN à s'adapter au monde industriel. A la différence des WLAN dont l'objectif principal, voire unique, est de permettre un accès sans fil à un réseau plus étendu, les WPAN ne permettent que de relier entre eux des équipements autonomes et, de fait, sont conçus pour s'adapter parfaitement au monde de l'embarqué.

### 1.5.1 IrDA

IrDA [IRDA] définit une norme de communication qui propose d'utiliser le médium infrarouge pour créer des réseaux de très courtes portées pour des débits allant jusqu'à 16 Mbits/s. IrDA spécifie de nombreux protocoles : IrPHY, IrLAP, IrLMP, IrCOMM, Tiny TP, IrOBEX, IrLAN et IrFR; grâce à cet ensemble complet de protocoles, IrDA se suffit à elle-même et peut fonctionner seule, sans autre protocole comme notamment les protocoles liés à TCP/IP. Nous verrons par la suite que cette spécificité est très répandue chez les réseaux de la famille des PAN.

Bien que la pile soit très complète, nous ne présenterons ici que les deux premières couches du modèle OSI, **IrPHY** (pour la couche physique) et **IrLAP** (pour la couche *liaison de données*).

- **IrPHY** spécifie la couche physique d'IrDA. Quatre couches physiques sont proposées : SIR (*Serial InfraRed*) propose de couvrir directement les débits normalisés dans le cadre d'une transmission RS232 classique (9600 bits/s, 19,2 kbits/s, 38,4 kbits/s, 57,6 kbits/s et 115,2 kbits/s). MIR (*Medium InfraRed*) propose deux débits plus élevés de 576 kbits/s et 1,152 Mbits/s, FIR (*Fast InfraRed*) propose un débit de 4 Mbits/s et enfin VFIR (*Very Fast InfraRed*) propose un débit de 16 Mbits/s. Une dernière version - UFIR (*Ultra Fast InfraRed*), 100 Mbits/s - est toujours en cours de développement.
- **IrLAP** est la couche liaison d'IrDA. Elle gère le contrôle d'accès au médium, la découverte automatique des stations potentiellement connectables; elle fiabilise la transmission et négocie les rôles *Primaire / Secondaire*. Selon la spécification IrDA, la station primaire (unique) contrôle l'accès au médium des stations secondaires<sup>7</sup>.

La principale caractéristique d'une transmission IrDA (qui en fait son originalité mais aussi son principal défaut) est que deux équipements IrDA doivent forcément être en ligne de vue (LOS, *Line Of Sight*) pour pouvoir communiquer ensemble. De plus, la plupart des transmetteurs IrDA sont assez directifs (environ 15°), ce qui oblige l'utilisateur à orienter les équipements et peut poser quelques problèmes. En revanche, à la différence de toutes les technologies utilisant le médium radio, un réseau IrDA ne peut traverser les murs, ce qui lui confère un certain atout quant à la sécurité et au cloisonnement

<sup>6</sup>Ironie du sort, 802.11s propose aujourd'hui ce qu'HiperLAN/1 proposait il y a plus de dix ans!

<sup>7</sup>Nous aurons maintes fois l'occasion de constater que cette technique est très répandue...

(réutilisation des canaux).

Depuis la fin des années 90, IrDA est largement déployée dans les ordinateurs portables et les téléphones cellulaires. Elle permet, entre autres, de synchroniser son agenda et ses contacts et de se servir du téléphone cellulaire comme modem. Cependant, l'apparition de Bluetooth (que nous verrons plus loin) fait que IrDA est désormais moins utilisée car son concurrent propose une plus grande portée et ne nécessite pas l'orientation des éléments émetteurs/récepteurs et une vue directe.

### 1.5.2 HomeRF

Comme son nom l'indique, HomeRF [LANS 99] est une norme de réseau destinée à un usage domestique pour partager un accès à Internet ou transporter des communications téléphoniques DECT<sup>◇</sup>. Elle a été imaginée par un consortium industriel formé entre autres par HP, IBM, Siemens, Proxim, Compaq, Intel et Microsoft mais elle souffre aujourd'hui de la concurrence des autres technologies WLAN, notamment depuis que Microsoft et Intel se sont éloignés du projet.

HomeRF proposait une couche physique travaillant dans la bande des 2,4 GHz, en FHSS (à 50 sauts par secondes), sur une modulation de type 2-FSK ou 4-FSK. Le débit bande de base est de 1 Mbits/s ou 2 Mbits/s, suivant la modulation utilisée. La portée typique est de l'ordre d'une cinquantaine de mètres, pour une puissance de 100 mW. HomeRF permet deux types de transports : asynchrones (typiquement pour de l'Internet via du TCP/IP) et synchrones (pour de la voix). De part son faible débit, HomeRF n'est pas adapté au transport de la vidéo. La capacité d'adressage est de 127 stations ; il est possible de multiplexer jusqu'à 6 liaisons voix à 32 kbits/s. L'accès au médium est réalisé selon le protocole SWAP, pour *Shared Wireless Access Protocol*, qui mélange CSMA (pour les données asynchrones) et TDMA (pour les données isochrones).

Bien que proposant une méthode d'accès au médium intéressante pour la cohabitation de flux asynchrones et synchrones, HomeRF n'a pas connue de succès et est abandonnée aujourd'hui.

### 1.5.3 IEEE 802.15.1, Bluetooth

IEEE 802.15.1/Bluetooth [IEE1 02] est une norme de réseau sans fil fédérateur dont l'idée est née au début des années 1990. Son objectif initial était de proposer une norme universelle pour les communications sans fil, plus performante et plus globale que les liaisons infrarouge, déjà très répandues à l'époque. Comme IEEE 802.11, Bluetooth s'est imposée grâce aux forts soutiens financiers dont elle a bénéficié. Aujourd'hui, elle se positionne essentiellement sur le marché des téléphones portables et accessoires associés, mais également de la bureautique, l'objectif étant de pouvoir facilement « connecter » un téléphone ou un ordinateur avec ses divers accessoires (oreillette, casque, souris, clavier, etc.). Les applications sont multiples et à l'heure actuelle, cette technologie profite de sa maturité. Son utilisation est généralement simple ce qui rend Bluetooth très populaire. De plus, à l'instar de 802.11 avec WiFi et de 802.16 avec WiMAX, Bluetooth est promue par un SIG (*Special Interest Group*).

La grande différence entre les deux très populaires technologies WiFi et Bluetooth est que cette dernière, comme IrDA, propose une pile protocolaire complète, de la couche physique à la couche application, à la différence de WiFi qui se cantonne aux niveaux 1 et 2 de la pile OSI en ne proposant ni plus ni moins qu'une version « sans fil » d'Ethernet.

Bluetooth utilise la bande ISM des 2,4 GHz (2,402 GHz à 2,480 GHz) en divisant cette bande en 79 canaux de largeur 1 MHz et met en œuvre une modulation de type GFSK<sup>8</sup> en FHSS. Le débit en bande de base est de 1 Mbits/s et la fréquence nominale des sauts FHSS de 1600 Hz, soit une période

---

<sup>8</sup>La GFSK est une modulation Gaussienne de fréquence. Les fréquences porteuses ne sont pas à valeurs discrètes comme en FSK mais varient dans le temps de manière à ne pas occasionner de discontinuités. D'un point de vue spectral, la GFSK est moins consommatrice que la FSK.



de  $625 \mu s$ . Du fait de l'utilisation des sauts de fréquence, Bluetooth impose un fonctionnement en maître/esclave pour le partage du médium radio.

Les topologies envisageables d'un réseau Bluetooth sont les suivantes : *point à point* (un maître, un esclave, figure 1.4), *Piconet* (un maître, deux à sept esclaves actifs, figure 1.5) ou *Scatternet* (jusqu'à 10 piconets reliés ensemble, figure 1.6). Quelque soit la topologie choisie, toutes les communications de station esclave à station esclave passent par la station maître ; les esclaves n'ont donc pas besoin d'être à portée radio les uns des autres. Bluetooth résout ainsi le problème classique en sans fil de *la station cachée*. Cependant, le cas particulier du *Scatternet* impose qu'un nœud puisse être membre de deux *Piconets* simultanément : tantôt maître de son *Piconet*, tantôt esclave d'un autre *Piconet*. Ces deux possibilités sont illustrées dans la figure 1.6.

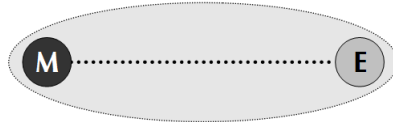


FIG. 1.4 – Topologie point à point de Bluetooth

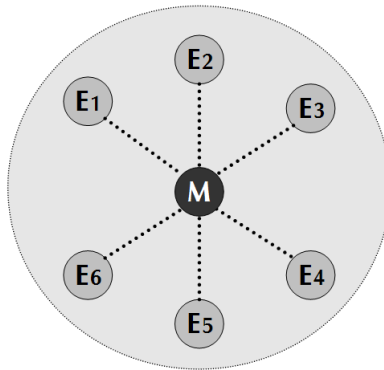


FIG. 1.5 – Topologie *Piconet* de Bluetooth

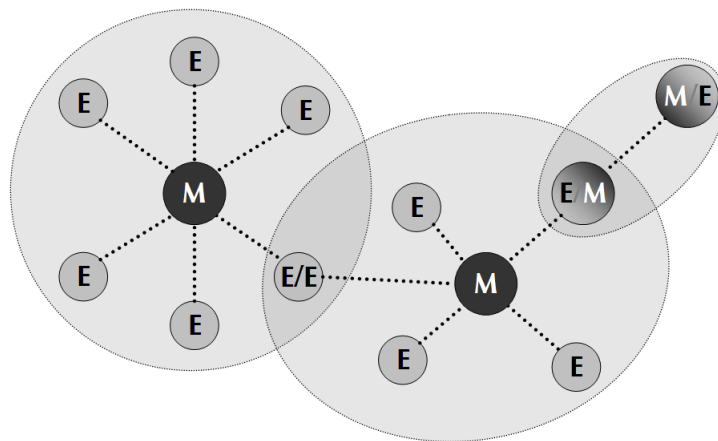


FIG. 1.6 – Topologie *Scatternet* de Bluetooth

Comme HomeRF, Bluetooth propose deux types distincts de transports : ACL (asynchrones, orientés « données ») et SCO (synchrones, orientés « voix »). Bluetooth propose également quelques éléments permettant la mise en œuvre de la Qualité de Service, comme nous le verrons dans la partie 3.3.4 de ce chapitre.

#### 1.5.4 IEEE 802.15.3, Ultra Wide Band (UWB)

Le standard IEEE 802.15.3 [IEE1 03] propose de normaliser une couche *physique* et une couche *liaison de données* basées sur le concept *Ultra Wide Band* (UWB). L'UWB est une technique d'étalement de spectre qui consiste à transmettre les données sur un spectre très large en un temps très court ; le signal ainsi transmis est proche d'une impulsion. La figure 1.7 illustre ce principe qui est également utilisé dans les technologies de radars.

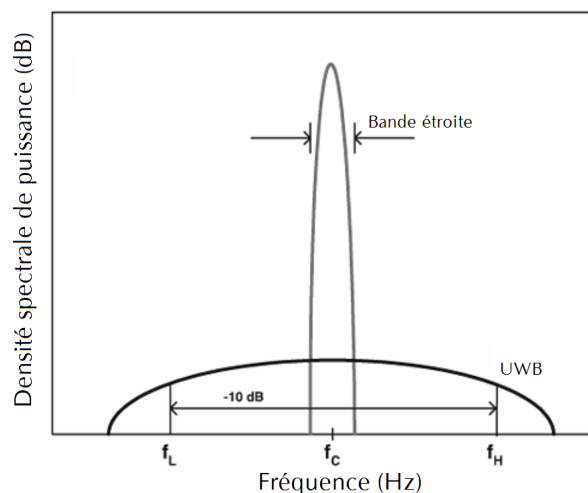


FIG. 1.7 – Le principe de l'Ultra Wide Band : utilisation du spectre radio fréquence

L'UWB, utilisé dans le cadre des transmissions de données, permet d'atteindre des débits très importants (près d'un Gbits/s), mais sur une distance assez courte. En effet, la bande passante d'un système de communication UWB décroît exponentiellement avec la distance ; de ce fait, un réseau UWB maillé à forte densité de nœuds devrait théoriquement proposer un débit plus élevé qu'un réseau UWB à faible densité de nœuds. L'UWB présente également des avantages au niveau propagation : de part la brièveté de l'émission, les transmissions basées sur UWB présentent une très bonne immunité aux multi-trajets.

Du fait de l'originalité de la méthode de modulation et de codage de l'information, le standard 802.15.3, notamment dans la méthode d'accès, devrait être amené à proposer des méthodes très innovantes. Bien que cela ne soit pas l'objectif du TG15.3 de l'IEEE, on peut également entrevoir de nombreux travaux de recherche au niveau du protocole de routage dans le cadre réseau maillé UWB : la décroissance exponentielle de la bande passante avec la distance imposant le principe *mieux vaut router que rechercher la portée la plus longue*, va à l'encontre des principes de routage actuels. Les perspectives à ce niveau semblent nombreuses.

Bien que UWB soit très prometteur, aucun produit répondant aux spécifications de 802.15.3 n'est encore disponible à ce jour. Cependant, le SIG Bluetooth prévoirait d'intégrer une couche physique de type UWB dans la troisième version de ses spécifications.

### 1.5.5 IEEE 802.15.4, ZigBee

Le TG15.4 de l'IEEE propose une norme pour les réseaux sans fil personnels à faible consommation énergétique (*Low Power-Wireless Personal Area Network*, LP-WPAN). Le standard IEEE 802.15.4 [IEE2 03] propose une norme pour les couches physique et liaison de données orientée très faible consommation énergétique qui rend cette technologie bien adaptée aux réseaux de capteurs. Nos travaux se basant sur 802.15.4, nous n'entrerons pas plus dans les détails de cette technologie qui sera largement présentée et étudiée dans les chapitres suivants.

## 1.6 Les réseaux de capteurs (WSN)

Les réseaux de capteurs (*Wireless Sensor Networks*) [CULL 04] constituent une catégorie de réseaux bien distincte des quatre familles vues jusqu'ici. Alors que les WWAN, WMAN, WLAN et WPAN sont conçus pour répondre à des problématiques de communications où l'homme est souvent un acteur principal (accès à un réseau global comme Internet, téléphonie, télécommande...), les WSN offrent des moyens de communication très souvent spontanés entre objets autonomes, généralement sans aucune intervention humaine.

Les réseaux de capteurs sont utilisés dans divers domaines :

- militaire : surveillance de zones tactiques, espionnage,
- environnement : surveillance de l'écosystème, prévention des risques sismiques,
- commerce : gestion de stocks, identification des colis,
- médical : assistance aux personnes,
- bâtiment : surveillance des infrastructures,
- transport : identification des bagages,
- ...

Bien que les WSN présentent des caractéristiques communes avec certains LP-WPAN (notamment au niveau des couches basses), les applications ne sont généralement pas les mêmes. D'une manière générale, les réseaux de capteurs se différencient des autres catégories de réseaux sur certaines caractéristiques [SIMP 05] :

- Les nœuds d'un WSN doivent pouvoir être déployés en très grand nombre (plusieurs centaines voire plusieurs milliers) sans difficulté. La phase de *passage à l'échelle* doit donc être validée pour un réseau composé de ce nombre de nœuds,
- Les nœuds d'un WSN n'ont pas de caractère individuel propre (on parlera de *population de nœuds*) et l'objectif de l'application a priorité sur la vie des nœuds (la gestion énergétique individuelle d'un nœud est moins importante que la gestion énergétique de l'ensemble du réseau),
- La portée typique d'un nœud WSN est faible (au maximum quelques dizaines de mètres); en revanche, un processus de routage est mis en œuvre pour permettre la communication globale sur tout le réseau, à la manière d'un réseau ad hoc,
- Un WSN est quasiment dédié à une unique application.

Au niveau des technologies, les WSN peuvent être déployés grâce à des nœuds passifs tels que les RFID ou sur des microsystèmes actifs. Plusieurs projets d'origine académique ou militaire ont donné lieu à des réalisations mais, à l'heure où ces lignes sont écrites, il ne semble pas qu'un standard ait normalisé précisément une technologie de WSN au même titre que les normes issues de l'IEEE ou de l'ETSI présentées jusqu'ici. Cependant, il semble que le standard de LP-WPAN IEEE 802.15.4 se rapproche très fortement des caractéristiques exigées pour un WSN.

## 2 La Qualité de Service dans les réseaux

Après avoir proposé un état de l'art des différentes technologies sans fil présentes sur le marché ou à l'étude, nous allons maintenant introduire le principe de *Qualité de Service* (QoS<sup>9</sup>) et l'illustrer, là encore, par l'existant. Nous constaterons à cette occasion que les travaux les plus récents liés à la *Qualité de Service* concernent majoritairement la couche 3 du modèle OSI (niveau *réseau*), même si nombre de travaux antérieurs concernaient la QoS au niveau *accès au médium*, notamment dans le domaine des Réseaux Locaux Industriels (RLI) [LEPA 91]. Nous reviendrons également sur l'intérêt de traiter la QoS dès les couches les plus basses.

### 2.1 Présentation générale

#### 2.1.1 Qu'est ce que la Qualité de Service ?

En réseau, la notion de Qualité de Service regroupe toutes les méthodes permettant de garantir la disponibilité du réseau pour ses utilisateurs. La notion de QoS repose sur l'hypothèse suivante : *le réseau possède des ressources finies et une pénurie de ses ressources est fortement possible dans des conditions d'utilisation raisonnables*. Le réseau étant utilisé simultanément par de nombreux utilisateurs dont les comportements individuels sont indépendants, ce risque de pénurie existe et il est tout à fait probable : cette pénurie serait provoquée par une demande d'utilisation massive et simultanée de la ressource partagée et aurait pour conséquence la congestion du réseau. Elle peut alors se traduire par un blocage des données et un effondrement partiel ou total des performances du réseau (augmentation de la latence de transmission, baisse du débit offert, pertes de paquets, etc.).

Dès lors, les bénéfices engendrés par une gestion fine de la QoS profitent aussi bien aux utilisateurs (meilleure disponibilité du réseau) qu'aux opérateurs (possibilité de facturer l'utilisateur en conséquence<sup>10</sup>!).

De ce fait, la Qualité de Service est un domaine de recherche et d'application vaste et prometteur car il vise à fiabiliser considérablement le réseau, d'éviter la pénurie des ressources et la dégradation de ses performances globales.

#### 2.1.2 QoS vs. Surdimensionnement

Nous pouvons opposer au concept de Qualité de Service la notion de *surdimensionnement*. Le surdimensionnement consiste à déployer un réseau dont la capacité est largement supérieure aux besoins réels, le risque de pénurie évoqué plus haut est donc minimisé mais toujours probable. Le surdimensionnement va très souvent de pair avec l'utilisation de protocoles basés sur le *Best-Effort*<sup>11</sup>, c'est-à-dire des protocoles ne proposant ni priorités ni garanties mais dont les performances sont acceptables dans la plupart des cas.

La mise en œuvre simple et le coût faible d'un réseau surdimensionné sont deux critères qui favorisent souvent l'utilisation de réseaux très haut débit au détriment d'un réseau à QoS, surtout dans des cas où le médium « ne coûte pas cher » comme dans le cas d'un réseau local filaire. Par exemple, le coût d'installation d'un réseau Ethernet Gigabit peut être très inférieur au coût d'équipements de contrôle de QoS. En revanche, dans d'autres cas où le médium peut avoir un *coût d'utilisation* plus élevé, l'utilisation

---

<sup>9</sup>L'acronyme *QoS* est très souvent utilisé pour désigner la notion de Qualité de Service.

<sup>10</sup>La notion de *tarification* est fondamentale lorsque le réseau public se voit doté de fonctionnalités de QoS : si l'on considère qu'en cas de pénurie, comme nous le verrons plus loin, certains trafics sont prioritaires devant d'autres, il est nécessaire de discipliner les utilisateurs, voire de les forcer à se discipliner en leur imposant un coût financier proportionnel à leur priorité sur le réseau. En revanche, dans le cas d'un réseau privé ou industriel, la QoS est administrée localement dans l'intérêt global, sans nécessité de surcoût.

<sup>11</sup>à traduire par *faire au mieux*.

de protocoles à QoS est rentable. Notons que la notion de « coût d'utilisation » n'est pas forcément d'ordre financier ; le médium peut être une ressource rare (réseaux sans fil hertziens, réseaux aquatiques acoustiques [BOUZ 03], réseaux à économie d'énergie) qu'il est important d'économiser.

Le réel inconvénient d'un réseau à QoS est qu'il nécessite une administration pointue pour être efficace. De plus, sur un réseau flexible comme un réseau *paquets*<sup>12</sup>, la mise en œuvre de la Qualité de Service affaiblit cette flexibilité et soulève des problématiques que l'on rencontre généralement sur un réseau *commuté*.

## 2.2 Notions fondamentales et principes de QoS

La mise en œuvre de la Qualité de Service sur un réseau implique certaines conditions. Celles-ci sont détaillées ci-après.

### 2.2.1 La prédictabilité du trafic

La mise en œuvre d'un réseau à Qualité de Service repose sur la prédictabilité du trafic. Pour pouvoir assurer une qualité constante quelqu'en soit la charge, l'administrateur doit être en mesure de pouvoir décrire précisément la physionomie de chaque trafic transporté par le réseau. Il est donc nécessaire d'évaluer les ressources demandées par les utilisateurs. On considère généralement qu'il n'est pas nécessaire pour l'application de savoir comment l'information est acheminée de bout en bout par le réseau ; cependant, l'utilisateur final est sensible à certaines propriétés qui ont un impact direct sur sa satisfaction vis-à-vis du réseau. On considère généralement que son ressenti peut se résumer par quatre paramètres qui sont :

1. la *latence*, c'est-à-dire le délai instantané qui impacte la transmission de bout en bout,
2. la *gigue*, c'est-à-dire la différence de latence entre les paquets (certains vont très vite, d'autres plus lentement),
3. le *débit*, c'est-à-dire le volume d'information transporté par unité de temps,
4. la *perte de paquets*, c'est-à-dire le taux de paquets qui n'arrivent pas à leur destination.

Suivant le type d'application visé, les quatre paramètres n'ont pas la même importance. Par exemple, dans le cadre d'un transfert de fichier, la latence et la gigue ont peu d'impact sur l'utilisateur, l'essentiel étant que le fichier soit transféré rapidement et sans erreur. Dans le cadre d'une communication téléphonique (transport multimédia interactif), le débit importe peu, tant qu'il suffit à transporter les données vocales numérisées ; cependant les paramètres temporels (latence, gigue) et le taux de paquets perdus ne doivent pas dépasser un certain seuil pour le confort des participants à la conversation téléphonique.

On peut alors définir deux paramètres pour évaluer les besoins d'une application communicante :

1. *L'élasticité*, c'est-à-dire la faculté d'une application réseau à s'adapter [MICH 05] à un changement brusque de qualité de connexion (typiquement, une application de transfert de fichier ou une application de *streaming vidéo* avec adaptation automatique du CODEC selon le débit de la ligne est fortement élastique),
2. *L'interactivité*, c'est-à-dire le niveau d'interaction ou de réactivité nécessaire pour un bon fonctionnement de l'application entre les deux points communicants (typiquement, les applications de type temps réel sont fortement interactives).

---

<sup>12</sup>On désigne souvent par *réseau paquets* un réseau à *commutation de paquets*, c'est à dire un réseau où les informations sont découpées en paquets de données pour être acheminées de bout en bout, par opposition aux *réseaux commutés* (réseaux à *commutation de circuits*) où le canal est totalement dédié à la transmission en cours. La paquetisation permet un multiplexage temporel qui autorise la réutilisation du médium en cas d'absence de données.

Une fois la physionomie du trafic clairement identifiée, des zones de tolérance peuvent être clairement énoncées et le trafic ne doit pas en sortir. Si c'est le cas, les paquets non conformes peuvent être refusés ou retardés par un reconditionnement (écrêtage du débit au niveau réseau ou transport). Éventuellement, dans des cas bien précis, le contenu peut être altéré (par exemple par une recompression à la volée de données multimédia).

### 2.2.2 Le contrôle d'admission

Un réseau à Qualité de Service ne peut pas s'engager à transporter tous les flux qui se présentent à lui ; la mise en application d'un protocole à QoS va obligatoirement de pair avec la mise en place d'un système de *contrôle d'admission*, c'est-à-dire un système qui agit en entrée du réseau et dont le rôle est d'accepter ou de refuser la prise en charge d'un nouveau trafic par le réseau. Un réseau à QoS connaît ses limites et refusera la prise en charge d'une nouvelle demande qui ne pourra pas être satisfaite dans les conditions demandées<sup>13</sup>.

### 2.2.3 Les degrés de Qualité de Service

Les trois principaux degrés de Qualité de Service (trois niveaux de services), du plus fiable au plus lâche, sont les suivants :

1. Le *service garanti*, ou *premium*. Il vise à émuler une liaison spécialisée : malgré un multiplexage des paquets sur le médium, le lien propose les mêmes garanties que s'il était basé sur une ligne indépendante. Des pertes de paquets ou une certaine gigue peuvent néanmoins être acceptées en fonction du contrat négocié. Au niveau technologies, le service garanti se retrouve avec le *GS (Guaranteed Service) de IntServ*, le *EF (Expedited Forwarding) de DiffServ* et le *CBR (Constant Bit Rate) de l'ATM* que nous détaillerons plus loin,
2. Le *service « mieux que Best-Effort »*.
3. Le *service Best-Effort*. Le protocole IP de base en est un exemple, ou encore UBR de l'ATM (*Unspecified Bit Rate*).

En pratique, l'application de la QoS sur le réseau se concrétise par deux types d'actions :

1. La **différenciation de services**. Dans le cadre de la différenciation de services, certains trafics sont traités prioritairement par rapport à d'autres : on parle de *traitement préférentiel* d'un paquet de données. La différenciation de services est réalisée d'une part grâce à un marquage des paquets (*tagging*) pour identifier leur niveau de priorité (ce marquage peut être réalisé au moment de l'admission dans le réseau) et d'autre part grâce à un ordonnancement (*scheduling*) au niveau du pré-traitement et de l'émission sur le réseau : dans les routeurs, les paquets de données ne seront pas traités selon le principe du « premier arrivé, premier servi » (FIFO, *First In, First Out*) mais selon leur priorité. La différenciation de services ne permet pas d'apporter de garantie absolue ; la seule garantie est relative aux autres trafics (plus prioritaire, moins prioritaire, ou même priorité).
2. La **réservation de ressources**. Dans le cadre de la réservation de ressources, il est possible de garantir l'acheminement des informations sur le réseau car une partie de ses ressources matérielles sera bloquée et dédiée à l'acheminement de cette information. Un *réseau paquets* avec réservation de ressources a une fiabilité proche de celle d'un *réseau à commutation de circuits*, le challenge étant de parvenir à une description physionomique du trafic la plus fidèle possible (proche possible de la réalité) pour ne réserver que les ressources nécessaires, et pas plus.

---

<sup>13</sup>Nous sommes bien aux antipodes du réseau *Best-Effort* !

## 2.3 Quelques exemples de protocoles à QoS

Nous allons ici présenter plusieurs exemples de protocoles implémentant une Qualité de Service. Nous allons le voir, les approches sont différentes mais les objectifs finaux sont les mêmes : permettre à l'utilisateur de disposer du réseau à tout instant, dans des conditions négociées au préalable et ce, quelque soit l'état de ce réseau à l'instant où les données devront être acheminées.

### 2.3.1 Asynchronous Transfer Mode (ATM)

ATM [PUJO 95], pour *Asynchronous Transfer Mode*, est une technologie Télécoms fonctionnant en mode connecté par commutation de cellules (*Cells*), établissant des circuits virtuels. L'objectif d'ATM est de pouvoir faire cohabiter des applications très diverses au sein d'un même réseau d'opérateur, avec une prise en charge native de la Qualité de Service : ATM est doté d'un protocole de signalisation qui négocie la QoS dès l'établissement du circuit virtuel [DESG 03]. Nous n'aborderons pas ici l'aspect routage par commutation de cellules mais simplement l'aspect QoS qui est intrinsèquement lié à la mise en œuvre d'ATM.

ATM propose une gestion du trafic par contrats, associés à des descripteurs de trafics et un mécanisme de contrôle d'accès et de priorité.

Comme nous l'avons vu plus haut, il est fondamental, dans un réseau à QoS, de savoir décrire le profil de chaque trafic écoulé par le réseau (débit continu ou en rafale, etc.). Afin de pouvoir réaliser cette description, la norme ATM propose six descripteurs de trafic :

- PCR (*Peak Cell Rate*)
- MCR (*Min Cell Rate*)
- SCR (*Sustainable Cell Rate*)
- CTD (*Cell Transfert Delay*)
- CDV (*Cell Delay Variation*)
- CLR (*Cell Loss Ratio*)

L'utilisation (ou non) de ces descripteurs de trafic donne existence à cinq catégories de services théoriques normalisées :

- CBR (*Constant Bit Rate*) utilise PCR, CTD, CDV et CLR. Les circuits virtuels CBR sont conçus pour les liens à débit constant et à fortes contraintes temporelles,
- RT-VBR (*Real Time-Variable Bit Rate*) utilise PCR, SCR, CTD, CDV et CLR. Comme CBR, les circuits virtuels RT-VBR sont utilisés pour les liens à fortes contraintes temporelles mais à débit variable,
- NRT-VBR (*Non Real Time-Variable Bit Rate*) utilise PCR, SCR et CLR. Dans le cadre de NRT-VBR, les descripteurs de trafic qui concernent les délais ne sont pas pris en compte. Les circuits virtuels NRT-VBR sont utilisés pour les liens à débit variable sans contraintes temporelles,
- ABR (*Available Bit Rate*) utilise PCR et MCR. Les circuits virtuels ABR ne tiennent compte que de deux bornes minimales et maximales pour le débit. Ils permettent de ne garantir qu'un débit minimal et un débit maximal.
- UBR (*Unspecified Bit Rate*) n'utilise aucun descripteur de trafic. Les circuits virtuels UBR ne permettent aucune garantie. C'est le mode *Best-Effort* de l'ATM.

Chaque service proposé par ATM est adapté à une utilisation particulière du réseau ; à chaque application communicante (transfert de fichiers, échange de données temps réel, multimédia, etc.) correspond un service adéquat. Quant aux deux derniers services (ABR et UBR), ils permettent d'utiliser les reli-

quats de bande passante inutilisés par les services à fortes contraintes. Ainsi, la partie de bande passante inutilisée qui avait été réservée par multiplexage statistique peut être revendue, ce qui est très intéressant et pour l'opérateur (revente de bande passante), et pour le client (accès au réseau sans garantie mais à moindre frais).

Lorsqu'un nouveau trafic entre dans le réseau ATM, il doit être accompagné de ses *descripteurs de trafic* et de son besoin en QoS (c'est-à-dire le niveau de QoS désiré par le client). Le nouveau trafic est alors soumis au CAC (*Call Admission Control*) qui prend la décision d'accepter le nouveau trafic dans le réseau, ou de le refuser, en fonction de la charge. Si le CAC est favorable, le trafic est découpé en cellules qui sont marquées (*tagging*) et injectées dans le réseau. A chaque passage dans un routeur du réseau, la conformité du trafic est vérifiée. Si le trafic n'est pas conforme à sa description, le surplus de cellules est détruit, entraînant une perte de données qui sera détectée par les couches supérieures<sup>14</sup>.

### 2.3.2 IntServ

IntServ [WROC 97] est un protocole de gestion de QoS pour un réseau de « petite taille ». Son nom est l'abréviation de INTEgrated SERVices et il est défini par les RFC 1633 et 2205-2216. L'objectif d'IntServ est d'éviter les problèmes de congestion dans les routeurs sur un réseau hétérogène non isochrone de bout en bout (comme Internet, par exemple). Pour cela, IntServ réserve une partie des ressources des routeurs du réseau en fixant des bornes sur les délais et les débits. Il spécifie un protocole de signalisation RSVP, pour *Ressource ReSerVation setup Protocol*, qui va être chargé de la communication entre routeurs à QoS. A la différence d'ATM qui fonctionne par l'émulation de circuits virtuels, un réseau comme Internet considère les paquets de manière indépendante (principe de la commutation de paquets). IntServ doit donc reconstituer un « flot » pour permettre le traitement de la Qualité de Service sur un ensemble de paquets qui seront identifiés comme faisant partie d'un même groupe qui doivent être traités de manière homogène d'un point de vue QoS.

#### 2.3.2.1 Profils de QoS au niveau utilisateur

Au niveau utilisateur, IntServ définit trois types de profils :

1. le service classique en *Best-Effort*,
2. un service de *charge contrôlée (Controlled Load)* où le réseau se comporte comme un réseau *Best-Effort* peu chargé : trafic interactif possible, débit moyen et rafale possible si l'utilisateur l'a demandé,
3. un service garanti (*Guaranteed load*) où l'utilisateur possède une « vraie » garantie sur le débit, le délai et la gigue.

De la même manière que pour tous les autres protocoles à QoS, un flot de données réclamant une QoS devra être accepté par un système de contrôle d'admission et devra par la suite être conforme à la physionomie décrite.

#### 2.3.2.2 Le protocole de signalisation RSVP

Le protocole RSVP est le protocole de signalisation spécifié par IntServ. Il permet aux routeurs de communiquer entre eux et d'établir la réservation de ressources de bout en bout, selon le contrat signé avec l'utilisateur. Les deux messages principaux de RSVP sont PATH et RESV. Le premier (PATH) est envoyé par un nœud désirant envoyer des données sur le réseau. Le message contient le descripteur de

---

<sup>14</sup>Cette méthode peut sembler brutale mais si le trafic n'est pas conforme à sa description préalable, le client n'a pas respecté son contrat...



Qualité de Service désirée. Chaque routeur du réseau recevant un message **PATH** retient l'adresse du routeur l'ayant déjà relayé. Lorsque le message **PATH** est reçu par le destinataire final, celui-ci répond en renvoyant un message **RESV** qui suit exactement même chemin, mais dans l'autre sens. Une fois le message **RESV** acheminé jusqu'à sa destination, le chemin est établi avec les réservations nécessaires. La figure 1.8 illustre le principe.

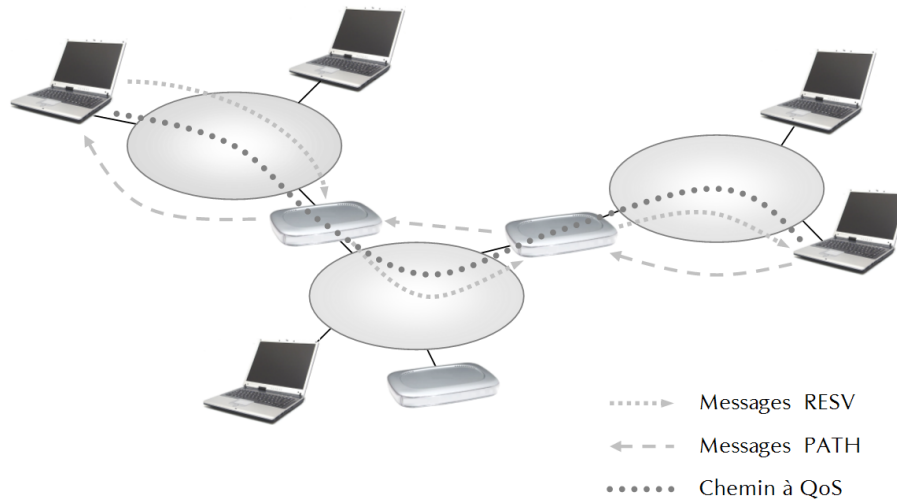


FIG. 1.8 – Illustration du protocole RSVP

Une fois le chemin établi, l'émetteur doit maintenir la réservation en reproduisant ce mécanisme à intervalles réguliers (réémission des messages **PATH** et **RESV**). Un changement de réservation (plus de débit, etc.) peut être effectué de la même manière. Si l'un des routeurs sur le chemin ne reçoit plus les messages au bout d'un certain temps, la réservation expire et les ressources sont libérées. Le protocole prévoit cependant que les réservations accumulées doivent être rendues par les demandeurs via les messages **PATH\_TEAR** et **RESV\_TEAR**. Ces deux messages sont utilisés de la même manière que **PATH** et **RESV**.

### 2.3.2.3 Conclusion : bénéfices et difficultés liés à IntServ

L'utilisation de IntServ et RSVP sur un réseau présente plusieurs avantages :

- IntServ permet la réservation de ressources dans les routeurs sans grandes modifications. Son intégration dans un réseau existant est relativement simple et son principe de traitement *par flots* s'adapte à tout réseau de type *paquets*.
- RSVP s'intègre parfaitement dans une architecture de réseau *multicast*, comme par exemple la diffusion de flux multimédia, où les principes de QoS sont très appréciés.

Cependant, IntServ présente aussi certains inconvénients :

- Le choix du traitement par flot est très consommateur de ressources : les tables de descripteurs de flots consomment beaucoup de mémoire, même pour un flot qui peut être très court : sur Internet, une requête HTTPv1.0 tient en quelques dizaines d'octets. Il sera plus long d'établir la QoS que de traiter la demande elle-même. De plus, si la taille du réseau et/ou le nombre de flots est grand, la quantité de mémoire nécessaire aux tables de descripteurs de flots devient trop importante. De ce fait, IntServ n'est pas adapté aux grands réseaux comme Internet,
- De part la création du chemin de réservation, IntServ n'est pas adapté à un réseau mobile sans fil

et revient à utiliser le réseau paquets comme un réseau à commutation de circuits. Une partie de la souplesse est perdue,

- Enfin, les messages RSVP étant transportés par le même protocole que les données utiles (signalisation et données transportées par le même réseau), ils risquent d'être perdus au même titre que les données en cas de congestion réseau ; la gestion de la QoS sera alors inefficace et les performances globales du réseau s'écrouleront. Idéalement, les messages de signalisation devraient être différenciés à un niveau plus bas (**Nécessité de la mise en œuvre d'une QoS au niveau Liaison**, comme nous le verrons à la fin de ce chapitre).

### 2.3.3 DiffServ

L'approche DiffServ [NIC1 98] [NIC2 98] (pour *DIFFerentiated SERVices*) est souvent comparée à celle de IntServ pour sa capacité à être déployée sur de « grands réseaux » ; alors que IntServ, de par le traitement par flots, ne peut s'appliquer que sur des réseaux de « petite taille », DiffServ réduit au maximum la taille des tables en considérant des agrégations de flots. Ainsi, les flots ne sont pas traités individuellement mais par agrégats, ce qui allège considérablement la charge des routeurs du réseau. De plus, le contrôle d'admission n'est plus assuré individuellement par chaque routeur traversé, mais par les *routeurs de bordures*, rendant DiffServ beaucoup plus adapté aux grands réseaux, et notamment aux *Réseaux d'opérateurs*<sup>15</sup>.

Le niveau de QoS d'un flot est indiqué dans un champ de l'en-tête de ses paquets. Lorsqu'un routeur de bordure décide d'admettre un nouveau trafic dans le réseau, il fixe la valeur du champ DSCP (**Differentiated Services CodePoint**) dans l'en-tête IP :

- le code 0 signifie que le paquet doit être traité en *Best-Effort*, après tous les autres (niveau le moins prioritaire),
- un code autre que 0 aura une autre signification ; cette correspondance est fixée par l'opérateur lui-même, selon les contrats qu'il propose à ses clients.

Lorsqu'un routeur du cœur du réseau (routeur « core ») devra traiter ce paquet, il devra inspecter le champ DSCP et traiter le paquet en conséquence. Ainsi, DiffServ présente l'avantage de ne pas nécessiter de signalisation puisque le décodage du champ DSCP se fait via des tables inscrites dans la mémoire du routeur. Le contrôle de congestion se fait directement par les routeurs de bordure, selon le principe de la capacité finie et connue du réseau. Dans le cœur du réseau, il n'y a pas de réservation de ressources, seule la différenciation de traitement des paquets suffit à appliquer une qualité de service pour la traversée du paquet dans le réseau. Cependant, afin de palier les pertes sur le médium, le réseau cœur, par exemple un réseau ATM, nécessite d'être légèrement surdimensionné.

## 2.4 Conclusion

Nous avons pris pour exemple trois protocoles qui mettent en œuvre de la QoS sur un réseau. Les protocoles sont différents mais la recette reste la même : description du trafic, acceptation si le réseau en a la capacité, acheminement de bout en bout. Éventuellement, le principe de différenciation de services permet d'acheminer les données les plus prioritaires en cas d'une éventuelle congestion.

Cependant, nous avons pu constater que la gestion de la QoS sur tout le réseau n'est pas chose simple. Une solution de type IntServ nécessite un trafic de gestion important (qui peut devenir très lourd pour des petits trafics irréguliers) et n'est pas du tout adaptée à un réseau mobile. Une solution de type DiffServ nécessite un surdimensionnement du réseau cœur pour palier aux pertes des couches 1 et 2, ce qui, en sans fil n'est pas envisageable (les pertes de niveau 1 et 2 sont trop importantes). Quelque

---

<sup>15</sup>Les réseaux d'opérateurs sont constitués de trois types de routeurs : les routeurs *core* qui constitue le cœur du réseau, les routeurs de bordures (*edge*) qui échangent les données avec d'autres réseaux (on parle aussi de *peering*) et les routeurs d'accès, qui font le lien avec les clients.

soit la solution retenue, il est essentiel de limiter les pertes à tous les niveaux car pour le système de contrôle d'admission, de fortes pertes aux niveaux 1 et 2 constituent une source d'incertitude qui lui font perdre sa pertinence ; pour la couche physique, un bit transmis correctement qui serait ensuite détruit par les couches supérieures pour lisser le trafic constitue une perte sèche de bande passante.

La solution idéale doit assurer une transversalité de la QoS dans toute la pile protocolaire.

### 3 Problématique de l'accès au médium partagé

Dans les techniques de télécommunications, un des problèmes fondamentaux est *le contrôle d'accès au médium partagé* : lorsque deux émetteurs utilisant le même canal de communication émettent des informations en même temps, il y a risque de perturbations pour les éventuels récepteurs. Il est possible de comprendre cette problématique en faisant une analogie avec la parole humaine : si dans une même pièce, deux personnes proches l'une de l'autre parlent en même temps, la conversation sera très difficile à suivre, voire inaudible pour eux-mêmes ou leurs interlocuteurs. Dans le cadre des télécommunications, le problème est exactement le même, qu'il s'agisse de télécommunications filaires ou sans fil. Dans la mesure où une perturbation risque de dégrader considérablement les performances du réseau de communication, on cherchera généralement à éviter que deux émetteurs émettent en même temps sur le même canal. Si cette perturbation ne peut être évitée, on désignera ce phénomène par le terme *collision*.

Pour résoudre cette problématique, les concepteurs ont été amenés à développer des *méthodes d'accès au médium*, c'est-à-dire des algorithmes permettant de minimiser voire de supprimer le phénomène des collisions. On parlera aussi de *contrôle d'accès au médium*, et plus souvent en anglais, de *Medium Access Control*, ou MAC. De nombreuses techniques ont été envisagées pour la résolution de cette problématique ; nous nous limiterons à présenter les plus importantes.

#### 3.1 Accéder au médium partagé : les principes de base

Le canal de communication est généralement utilisé par plusieurs éléments communicants du réseau car il est doté d'une certaine capacité. Il suffit de coordonner la répartition de cette capacité pour faire cohabiter plusieurs émetteurs sur le même canal de communication sans qu'ils se perturbent et que leurs données émises n'entrent en collision. On parlera dans ce cas, d'accès multiple au médium ou de multiplexage, comme l'illustre la figure 1.9.

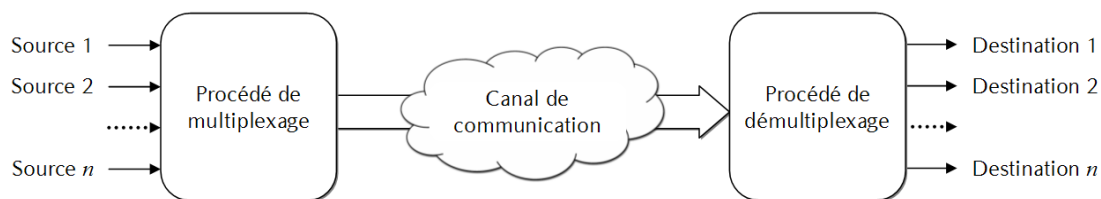


FIG. 1.9 – Procédé de multiplexage / démultiplexage pour la cohabitation de plusieurs trafics sur le même canal de communication

Le multiplexage consiste à diviser une composante du canal pour le répartir sur  $n$  émetteurs. En télécommunications, on considère qu'il existe trois types de multiplexage de base (trois *espaces* du canal divisibles) :

1. Un multiplexage de type *fréquentiel*, ou FDMA<sup>◇</sup> (*Frequency Division Multiple Access*) dans lequel l'espace fréquentiel est divisé entre tous les émetteurs. L'émission peut alors se faire temporellement en continu. Le FDMA est simple à mettre en œuvre mais son inconvénient principal est qu'il faut un récepteur dédié par canal à écouter. Cette technique de multiplexage est la plus ancienne ;

elle est encore largement utilisée dans les diffusions comme la télévision et la radio.

2. Un multiplexage de type *temporel*, ou TDMA<sup>◇</sup> (*Time Division Multiple Access*) dans lequel l'espace temporel est divisé entre tous les émetteurs. L'émission peut alors se faire sur la même bande de fréquence puisque celle-ci est tour à tour utilisée par les différents émetteurs du réseau. La méthode du TDMA est intéressante dans la mesure où il n'est pas nécessaire de changer de fréquence pour écouter d'autres émetteurs. En revanche, les émetteurs doivent se synchroniser pour ne pas utiliser le canal en même temps. Le TDMA est très adapté aux transports d'informations de type *paquets* ; c'est aussi pour cette raison que son principe est largement utilisé.
3. Un multiplexage par *codage*, ou CDMA<sup>◇</sup> (*Code Division Multiple Access*) dans lequel l'espace des codes est divisé entre tous les émetteurs. Chaque émetteur possède un code unique que ses récepteurs doivent connaître. Dans l'idéal, les codes choisis doivent être orthogonaux, c'est à dire qu'ils doivent respecter des propriétés mathématiques qui les rendent « totalement différents » y compris d'un point de vue spectral ; dans ce cas, si un récepteur ne connaît pas le code d'un émetteur, son signal sera reçu comme du bruit. Le CDMA, en tant que méthode d'accès, est assez peu utilisé dans la pratique, bien que le principe de l'étalement de spectre sur lequel il repose, soit, lui, largement utilisé, mais plutôt pour ses propriétés de fiabilisation de la transmission.

Bien entendu, ces trois multiplexages « de base » peuvent être combinés pour donner lieu à des techniques de multiplexages élaborées comme le FHSS, qui associe TDMA et FDMA en réalisant des sauts de fréquences. Plus simplement, d'autres techniques de diffusion comme le DVB-T<sup>◇</sup> (utilisé pour la Télévision Numérique Terrestre, TNT) ou le DVB-S (pour les Radios, Télévisions et services de données par satellite géostationnaire) prévoient d'émettre sur plusieurs fréquences (FDMA) et multiplexer plusieurs programmes sur une même fréquence (TDMA). On pourra également noter le cas particulier de l'OFDMA<sup>◇</sup>, basé sur le principe de l'OFDM, qui permet à plusieurs stations d'émettre en même temps sur des sous-porteuses différentes (un peu comme en FDMA) mais en ne nécessitant qu'un seul récepteur capable de recevoir simultanément toutes les sous-porteuses.

Cependant, dans le cadre du TDMA, le temps est divisé entre tous les émetteurs et il est nécessaire de répartir ce temps, si possible, de manière automatique et flexible. Bien que le concept de TDMA soit très simple à envisager, sa mise en œuvre, en revanche, n'est pas chose aisée. Nous allons voir par la suite plusieurs techniques d'accès au médium implémentées dans les technologies présentées en début de chapitre. Nous allons également constater que la plupart de ces méthodes d'accès sont basées sur des protocoles aléatoires qui ne règlent pas *de manière certaine* le problème des collisions ce qui, d'un point de vue QoS, peut poser problème.

## 3.2 Les principales techniques d'accès au médium

### 3.2.1 CSMA/CA (IEEE 802.11 DCF)

IEEE 802.11 [IEE1 99] propose deux méthodes d'accès au médium : la méthode DCF (*Distributed Coordination Function*) et la méthode PCF (*Point Coordination Function*) que nous présentons dans la section suivante.

La méthode d'accès DCF est une méthode d'accès en mode *Best-Effort*, c'est à dire sans priorité et sans garantie. Elle est très efficace pour transporter les trafics ne nécessitant pas de garantie sur la latence de transmission ou le débit offert (téléchargement, navigation sur Internet, etc.). Elle s'appuie sur l'algorithme CSMA/CA, qui est l'une des nombreuses variantes de la méthode d'accès au médium CSMA. Cette variante présente la particularité d'être adaptée au médium sans fil dans la mesure où, dans le cas d'un médium sans fil, il est difficile de détecter une collision instantanément. Contrairement à un médium filaire, il n'est pas possible, sur un médium sans fil, d'entendre le médium en même temps

que l'on transmet des données, ceci à cause des phénomènes d'*aveuglement*<sup>16</sup> et d'*affaiblissement*<sup>17</sup>, alors qu'en filaire, Ethernet utilise la variante CSMA/CD<sup>◇</sup> qui permet de détecter une collision quasi-instantanément : la comparaison de ce qui est émis et de ce qui est entendu permet de savoir si il y a collision alors que dans le cadre de CSMA/CA, la collision est détectée ultérieurement par la non-réception d'un message d'acquittement.

L'algorithme CSMA/CA fonctionne ainsi : une station voulant émettre des données commence par déterminer si le médium est libre :

- si c'est le cas, elle peut envoyer ses données immédiatement (figure 1.10),

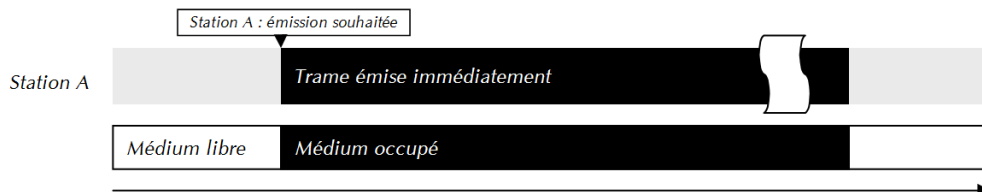


FIG. 1.10 – Illustration de CSMA/CA si le médium est initialement libre

- si le médium est occupé (figure 1.11), elle doit différer cet envoi de la manière suivante :
  - attente (1-persistent) que le médium redevienne libre,
  - puis attente d'un temps DIFS<sup>◇</sup> + un temps aléatoire de CW slots (*Contention Window*, fenêtre de contention<sup>18</sup>),
  - si, à la suite de cette attente, le médium est toujours libre, la station a gagné l'accès au médium et peut ainsi émettre ses données sans attendre plus longtemps (cas de la station A sur la figure 1.11),
  - si, alors qu'elle est dans la phase de fenêtre de contention, le médium est repris par une autre station, le processus est abandonné jusqu'à la prochaine période. Cependant, la station mémorise le temps qu'il lui restait à attendre, pour reprendre là où elle était restée, lors de la prochaine négociation d'accès au médium (cas de la station B sur la figure 1.11).

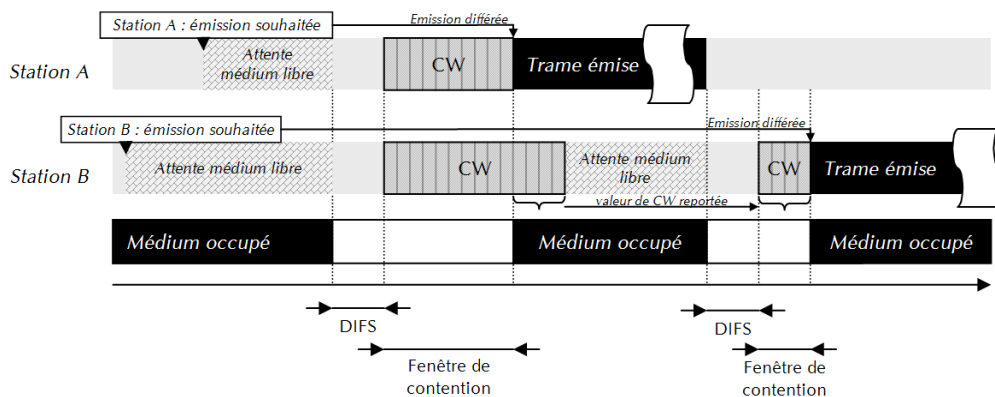


FIG. 1.11 – Illustration de CSMA/CA si le médium est initialement occupé

<sup>16</sup>Pour un récepteur recevant deux émetteurs simultanément, si la différence de puissance des signaux perçus est importante, le récepteur est dit *aveuglé* par le signal le plus puissant car il est dans l'impossibilité de décoder le signal le plus faible.

<sup>17</sup>En sans fil, une grande partie de l'énergie émise est perdue dès que le récepteur s'éloigne de l'émetteur. Cette propriété est démontrée par la formule de Friis

<sup>18</sup>Cette fenêtre de contention permet de différer la reprise de l'émission et, ainsi, de répartir l'accès au médium entre les stations.

Notons que les acquittements ne sont pas représentés sur les figures 1.10 et 1.11, et que, bien entendu, l'équité de cet algorithme repose sur le respect du tirage aléatoire de CW. Notons également qu'une station qui n'est pas en émission doit écouter le médium car si elle reçoit une trame qui lui est destinée, elle doit l'acquitter par l'envoi d'une trame d'acquiescement (ACK).

En ajoutant le mécanisme RTS/CTS à la technique CSMA/CA, IEEE 802.11 permet de contrer l'effet du terminal caché. Considérons trois stations A, B et C : B peut communiquer avec A et C, mais A et C ne peuvent communiquer ensemble, par exemple parce qu'elles sont trop éloignées l'une de l'autre ou parce qu'un obstacle les sépare. Considérons que A est en train d'émettre un message destiné à B, et que C désire également accéder au médium ; si C respecte le protocole CSMA/CA tel qu'il a été exposé plus haut, C va croire que le médium est libre et va pouvoir émettre immédiatement et à coup sûr provoquer une collision au niveau de B puisqu'elle n'entend pas A.

Pour contrer ce phénomène très fréquent en sans fil, avant de pouvoir envoyer ses données (trame DATA), une station devra d'abord envoyer une trame RTS (*Request To Send*) contenant l'adresse de la station de destination et la durée approximative de l'émission (NAV, *Network Allocation Vector*), jusqu'à l'acquiescement ACK ; la station de destination répond à ce message par un CTS (*Clear To Send*), en diffusant elle aussi l'information de durée contenue dans le RTS. Ainsi, toutes les stations à portée de la source comme de la destination sont informées qu'elles ne doivent pas utiliser le médium pendant le temps annoncé dans les trames RTS et CTS. La figure 1.12 illustre un réseau composé de trois stations A, B et C dont deux (A et C) ne se voient pas à cause d'un obstacle.

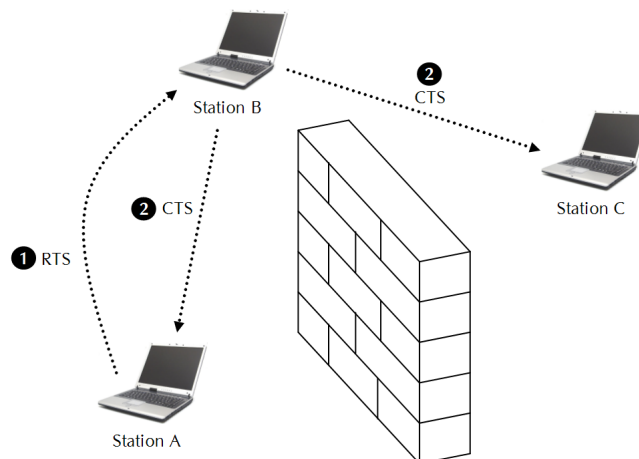


FIG. 1.12 – Topologie présentant deux terminaux mutuellement cachés

Au niveau de l'accès au médium, la figure 1.13 illustre le déroulement temporel de l'activité du médium. Dans cet exemple, nous avons ajouté une station D, qui ne reçoit que la station A.

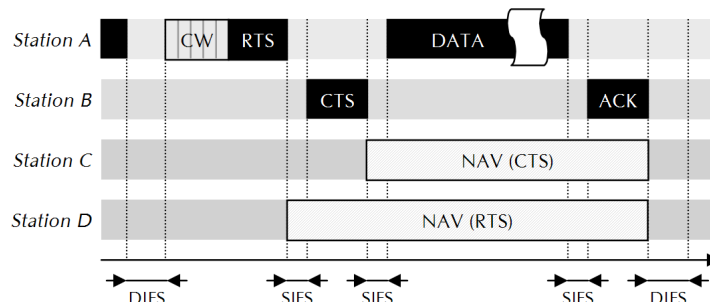


FIG. 1.13 – Déroulement de l'accès au médium par la méthode CSMA/CA

Notons que ce système ne permet pas d'éviter toutes les collisions, notamment les collisions de RTS et/ou CTS. A ce titre, 802.11 prévoit un seuil RTS-THRESHOLD sur la longueur de la trame à partir de laquelle le mécanisme de RTS/CTS est actif. Si la longueur de la trame est inférieure à ce seuil, le mécanisme n'est pas activé et la trame peut être envoyée directement. En effet, pour de petites trames, il y a autant de chance de provoquer une collision sur le RTS que sur la trame de donnée, donc autant envoyer la donnée directement sans perte de temps et de débit<sup>19</sup> [XU 02].

Pour finir cette description de 802.11 DCF, et comme on peut le voir sur la figure 1.13, 802.11 préconise l'usage de plusieurs temporisations inter-trames ; il en existe trois, de la plus courte à la plus longue :

1. **SIFS**, pour *Short Inter-Frame Spacing*, la plus courte, précède toutes les trames au sein d'un même échange, i.e. entre le RTS et le CTS, entre le CTS et les données puis enfin entre les données et l'accusé de réception : une fois l'accès au médium remporté, tout l'échange se fait avec des temps inter-trame prioritaires pour conserver l'accès au médium. Les trames émises avec un temps inter-trame SIFS seront donc les plus prioritaires,
2. **PIFS**, pour *Point Coordination Inter-Frame Spacing*, précède les trames de polling émises par le point d'accès en mode PCF (voir section 3.3.2),
3. **DIFS**, pour *Distributed Inter-Frame Spacing*, la plus longue, précède les trames envoyées en mode DCF. Comme cette durée inter-trame est la plus longue, on peut noter ici que les trames DCF sont moins prioritaires que les trames PCF.

Là encore, il est important de respecter ces trois temps pour conserver un accès au médium équitable.

Comme nous l'avions énoncé en l'introduisant, la méthode DCF proposée par l'IEEE pour 802.11 ne propose qu'un service de type *Best Effort*. En effet, même si le dispositif RTS/CTS permet d'éviter un grand nombre de collisions<sup>20</sup>, certaines sont inévitables, en particulier quand les deux stations pensent en même temps que le médium est libre. Cette méthode d'accès ne présente donc aucune garantie, ni sur le débit, ni sur la latence. En cas de trafic élevé, les performances du réseau s'écroulent<sup>21</sup>.

### 3.2.2 EY-NPMA (HiperLAN/1)

La méthode d'accès au médium proposée par l'ETSI [ETSI] pour HiperLAN/1 est le EY-NPMA<sup>◇</sup> (*Elimination Yield – Non-preemptive Priority Multiple Access*). Cette technique d'accès au médium [APOS 96] [WEIN 97] limite les collisions en les regroupant en début de trame et en les détectant, de manière à les éviter dans la suite de la transmission, jusqu'à la trame suivante. L'accès au médium est réalisé en trois étapes :

1. *Résolution des priorités* : le mécanisme d'accès tente d'attribuer le médium à la station la plus prioritaire : cinq niveaux de priorité sont prévus, de 0 (le plus prioritaire) à 4 (le moins prioritaire). Chaque nouvelle période de contention débute par 5 slots, un pour chaque niveau de priorité. Une station de priorité  $n$  n'a le droit de prendre la parole qu'à partir du slot  $n$  et seulement si aucune autre station plus prioritaire n'a pris la parole avant.
2. *Phase d'élimination* : une station voulant prendre le médium durant la phase de résolution des priorités émet alors un flot de données quelconque (*burst*) de durée aléatoire. Cette durée est

---

<sup>19</sup>Concernant des petites trames de données, le RTS/CTS constitue une perte de temps et de bande passante non négligeable!

<sup>20</sup>Notamment dans le cadre d'un réseau de type infrastructure, puisque toutes stations connectées au point d'accès entendent forcément le RTS ou le CTS, puisque la communication de station à station est impossible.

<sup>21</sup>Bien qu'il ne s'agisse pas d'une problématique de niveau MAC, nous pouvons également relever ici l'aberration de 802.11 qui oblige l'ensemble des stations d'un même réseau à travailler avec le débit le plus faiblement négocié. En effet, si toutes les stations négocient un débit de 11 Mbit/s alors qu'une seule station ne peut négocier un débit supérieur à 1 Mbit/s, l'ensemble du réseau travaillera à 1 Mbit/s, ce qui dégrade fortement les performances de l'ensemble du réseau.

distribuée selon une loi de probabilités discrètes. Toutes les stations de priorité  $n$  émettent ce flot de données quelconques.

3. *Phase de capitulation* : une fois cette durée aléatoire écoulée, la station passe à l'écoute : si aucune autre station n'est en train d'émettre, cela signifie qu'elle a gagné l'accès au médium. Elle peut alors émettre ses données. Si une autre station est en train d'émettre, elle « capitule » et ne transmettra pas ses données pendant cette période ; elle devra retenter sa chance à la période suivante.

La figure 1.14 illustre le déroulement d'un accès au médium réussi par EY-NPMA. Dans le cas illustré, le réseau est composé de trois stations, dont deux ont le même niveau de priorité. Au final, dans le cas illustré, c'est la station n°1 qui remporte l'accès au médium. En revanche, si deux stations de même priorité choisissent la même durée aléatoire, la phase de capitulation ne permettra pas de détecter la station concurrente et une collision sera produite. La figure 1.15 illustre ce cas.

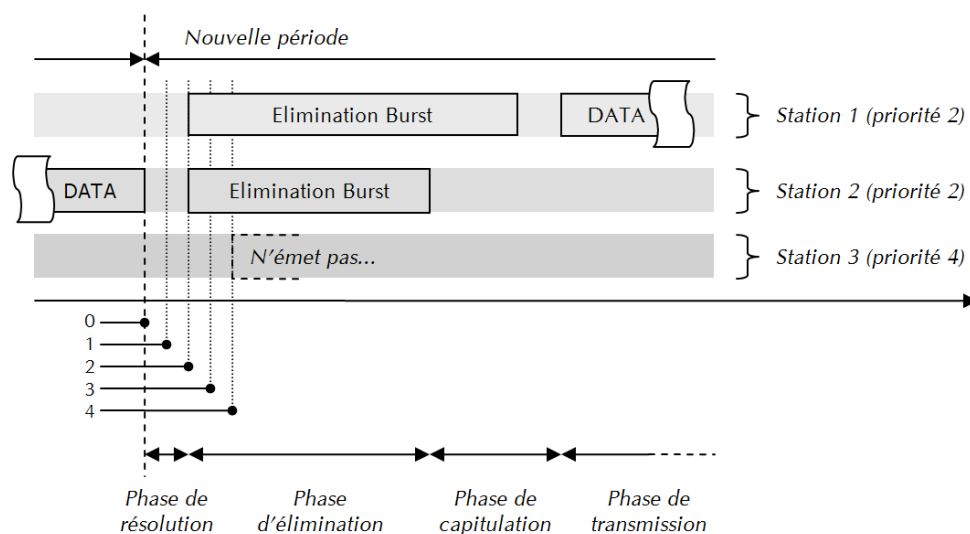


FIG. 1.14 – Déroulement d'un accès au médium réussi par la méthode EY-NPMA

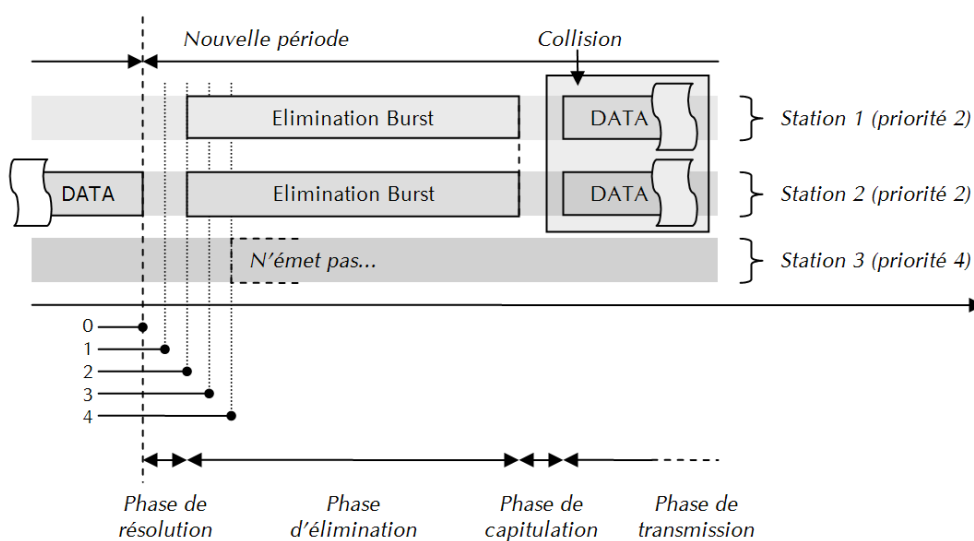


FIG. 1.15 – Déroulement d'un accès au médium par la méthode EY-NPMA ayant échoué



Notons cependant que cette méthode d'accès a un inconvénient majeur : elle ne règle pas le problème de la station cachée.

Dans le cadre de HiperLAN/1, l'ETSI propose donc la mise en œuvre d'une méthode d'accès qui règle le problème des collisions en se basant sur des grandeurs aléatoires, mais en permettant à certains flux ou certaines stations d'être prioritaires devant d'autres. Cette notion de *différenciation de service* que nous avons abordé plus haut est un premier pas vers l'intégration de la *Qualité de Service* dans le réseau. Cependant, et c'est là l'objectif de nos travaux, cette méthode d'accès n'offre aucune garantie sur le plan temporel, puisque la méthode d'accès n'est pas déterministe : en effet, si deux stations de même priorité tirent la même période aléatoire dans la phase d'élimination, il y a collision, comme l'illustre sur la figure 1.15. De plus, la non résolution du problème de la station cachée augmente considérablement la probabilité de collision.

### 3.3 Introduire de la QoS au niveau accès au médium

Les méthodes d'accès que nous venons de présenter ne permettent pas de pratiquer des accès au médium à QoS, même si la couche immédiatement supérieure le prévoit. 802.11 DCF ne propose qu'un service de type *Best-Effort* et au mieux, HiperLAN/1, avec EY-NPMA, ne permet que de rendre prioritaire certains services par rapport à d'autres. Mais aucune méthode d'accès ne propose de garantie stricte, c'est à dire un débit minimal et une latence de transmission maximale.

Dans cette partie, nous allons présenter des méthodes d'accès au médium qui proposent des fonctionnalités supplémentaires au niveau QoS ; nous verrons également que toutes ont cependant des limites dans ce domaine.

#### 3.3.1 TDMA Dynamique (HiperLAN/2)

La méthode d'accès au médium proposée pour HiperLAN/2 [ETSI 00] est le TDMA Dynamique [APOS 96]. Rappelons que HiperLAN/2 est un réseau *infrastructuré*, c'est à dire constitué de points d'accès (AP) qui permettent à des terminaux sans fil d'accéder à un réseau filaire. Les terminaux sont associés à l'un des points d'accès, l'accès au médium est cadencé par l'AP.

Dans HiperLAN/2, les transports se font en mode connecté ; chaque demande de connexion est effectuée par un terminal qui communique ses besoins à l'AP. Pour ce faire, le point d'accès divise le temps en portions appelées « supertrame ». A l'intérieur de la supertrame, le temps est encore divisé en quatre canaux temporels, comme illustré par la figure 1.16 :

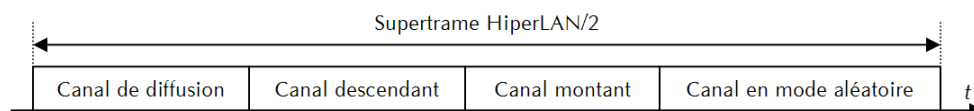


FIG. 1.16 – La supertrame HiperLAN/2

1. *canal de diffusion (broadcast channel)* : il contient des informations destinées à tous les terminaux comme sa puissance d'émission, son identifiant et l'identifiant du réseau, la structure des canaux descendant et montant (ressources accordées), les données diffusées, etc.
2. *canal descendant (downlink channel)* : il contient les données transportées dans le sens « point d'accès vers stations » selon la répartition décrite dans le canal de diffusion. Les données sont découpées en PDU (*Protocol Data Unit*) de 54 octets<sup>22</sup>.

<sup>22</sup>Ce découpage permet le transport direct et natif des cellules ATM par le réseau HiperLAN. Si c'était à refaire, cette décision ne serait peut-être plus prise aujourd'hui, car le très populaire IP domine au niveau des protocoles de niveau 3...

3. *canal montant (uplink channel)* : idem canal descendant mais dans le sens « terminaux vers point d'accès ». L'accès au médium se fait alors sans collision.
4. *canal d'accès aléatoire (random access channel)* : cette période est accordée par le point d'accès pour que les terminaux puissent effectuer les demandes de ressources pour les trames suivantes. Dans ce créneau temporel, l'accès au médium se fait en mode aléatoire.

Pour s'associer au réseau, un terminal HiperLAN/2 doit s'identifier auprès du point d'accès par l'envoi d'un message dans le *canal d'accès aléatoire*. Une fois l'accès accordé et la communication établie par le point d'accès, celui-ci accorde des slots temporels à ce nouveau terminal, en fonction des besoins qu'il a communiqué. Le transport des données se fait alors sans collision dans le canal montant. Les données à destination des stations connectées sont quant à elles transportées dans le canal descendant.

Le TDMA Dynamique proposé pour HiperLAN/2 limite donc fortement les collisions, au moins dans les phases *downlink* et *uplink channel*. Cependant, sur le plan temporel, cette méthode d'accès n'assure pas de garantie quant à la latence de transmission qui impacte les données transmises car la demande de connexion au réseau est effectuée en mode aléatoire. Nous verrons par la suite que ce défaut est très récurrent dans le monde des réseaux sans fil. Ce problème sera d'ailleurs évoqué dans le deuxième chapitre pour le cas d'IEEE 802.15.4. En revanche, une fois la connexion établie, un terminal est certain d'accéder régulièrement au médium, ce qui, mis à part la latence engendrée par les erreurs de transmission, permet de proposer une garantie de niveau liaison sur le débit offert et la latence. On peut d'ailleurs noter ici qu'HiperLAN/2 applique de fait le vieux principe adopté en télécommunications qui consiste à *assurer la qualité des communications déjà engagées au détriment d'éventuelles nouvelles connexions*.

### 3.3.2 Point Coordination Function (IEEE 802.11 PCF)

Pour palier aux lacunes de la méthode d'accès DCF, l'IEEE a initialement prévu une seconde méthode d'accès au médium pour 802.11, la méthode PCF (*Point Coordination Function*). Elle est basée sur l'interrogation des stations (*Polling*) par une entité spéciale appelée *point de coordination*. Ce rôle sera joué par le point d'accès. La méthode d'accès PCF permet de garantir l'accès au médium pour certaines stations, par exemple pour la transmission de données avec contraintes temporelles et/ou de débit comme la téléphonie ou la vidéo. Notons que, de par sa structure centralisée autour du point d'accès, la méthode d'accès PCF n'est pas utilisable sur un réseau de type ad hoc.

Dans le cadre de la méthode d'accès PCF, le point de coordination orchestre les accès au médium en interrogeant tour à tour les stations qui nécessitent une certaine garantie d'accès au médium. Cette méthode, proche de l'accès basique de type TDMA, permet d'introduire des priorités pour ces stations par rapport aux stations « normales ». Avec l'accès PCF, le temps est divisé en *supertrame*, permettant une alternance cyclique entre accès sans contention PCF et accès avec contention en DCF<sup>23</sup>. La partie avec accès PCF sera appelée CFP, pour *Contention Free Period*, alors que la partie DCF sera la CP, pour *Contention Period*. Le déroulement temporel d'une CFP est illustré figure 1.17 ; il se déroule ainsi :

1. Au début de la supertrame, le point de coordination sonde le canal. Si le canal est libre, le point de coordination attend une période PIFS et transmet une trame « balise », ou *beacon*. Cette trame marque le début de la supertrame et de la période d'accès en PCF. Elle contient aussi l'identifiant du réseau (BSSID, *Basic Service Set Identifier*), la période des trames *Beacon* et la durée maximale de la CFP. Ces deux dernières données vont permettre aux stations qui minimisent leur énergie de couper leurs fonctions de réception radio.
2. Ensuite, le point d'accès interroge tour à tour les stations (*polling*). Cette interrogation se fait par l'envoi de trames *CF-Pool* par le point de coordination si celui-ci n'a pas de données à envoyer à cette station, ou *DATA+CF-Pool* si des données du point d'accès sont en attente pour la station

---

<sup>23</sup>Une possibilité récurrente d'accès en DCF est nécessaire pour permettre aux stations non contraintes temporellement de pouvoir accéder au médium sans garantie en mode Best-Effort.

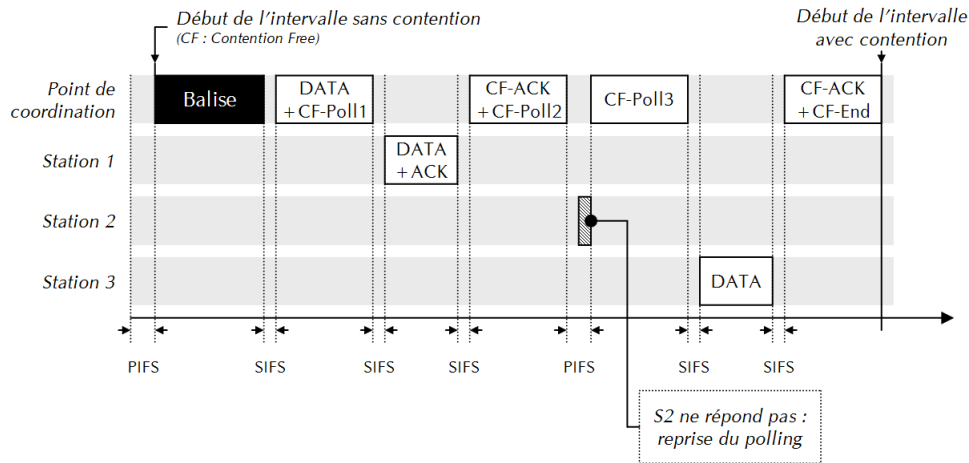


FIG. 1.17 – Déroulement temporel d'une période d'accès sans contention dans le cadre de 802.11 PCF

concernée. Le destinataire, quel qu'il soit, doit acquitter les données par une trame **CF-ACK**. Si la station interrogée doit envoyer des données, elle peut le faire après réception du **CF-Poll** ou du **DATA+CF-Poll**, en répondant respectivement par un **DATA** ou un **DATA+ACK**. Notons que tous ces échanges se font en utilisant un temps inter-trame SIFS, sauf si une station ne répond pas à la trame **CF-Poll**. Dans ce cas précis, le point de coordination reprend le médium après une durée PIFS pour continuer le polling. En tout état de cause, SIFS et PIFS étant plus courts que DIFS, les tentatives d'accès DCF pendant la CFP sont mécaniquement différés.

3. Pour finir, le point d'accès envoie une trame **CF-End** pour indiquer la fin de la période sans contention et, par conséquent, le début de la période avec contention.

De notre point de vue concernant le transport d'informations à fortes contraintes temporelles, la méthode d'accès PCF présente un avantage sur la méthode DCF : l'accès au médium des stations PCF n'est pas basé sur un algorithme aléatoire, ce qui permet d'entrevoir un embryon de gestion d'accès au médium avec Qualité de Service. Cependant, deux problèmes subsistent :

1. En premier lieu, pour figurer dans la liste des nœuds interrogés, les stations doivent être *associées* au point d'accès. Or comme c'est le cas pour beaucoup d'autres technologies WLAN/WPAN, le mécanisme d'association ne peut être considéré comme déterministe car la demande d'association est envoyée dans la DCF par utilisation de CSMA/CA. Si le médium est alors surchargé, l'association sera impossible, même si le nœud voulant entrer dans le réseau a des informations critiques à envoyer.
2. Enfin, les stations ayant souscrits à l'accès PCF ne sont pas pour autant certaines d'accéder au médium car le point d'accès peut différer l'interrogation d'une station si le temps restant avant la fin de la CFP n'est pas suffisant pour finaliser la transaction avec cette station. Ainsi, le mode d'accès PCF présente une trop forte caractéristique *égalitaire* dans la mesure où toutes les stations sont inlassablement interrogées à tour de rôle sans différenciation des besoins. De ce fait, le mode PCF ne permet pas d'envisager de garanties fortes sur le plan temporel.

### 3.3.3 Enhanced DCF et Hybrid CF (IEEE 802.11e)

Nous l'avons vu plus haut, la norme originale 802.11 ne propose pas d'outils réels pour la gestion de la qualité de service. Le mode DCF ne supporte que des trafics en *Best-Effort* et ne propose pas d'outils pour réserver la bande passante, ni pour limiter la latence et la gigue de transmission. De plus, les

nombreuses études réalisées montrent que les performances sont fortement dégradées [WEI 02] lorsque le réseau est très sollicité (nombre de nœuds, transferts nombreux et consommateurs de bande passante, modèles de trafics différents). Le mode PCF, bien qu'il propose théoriquement quelques améliorations, ne permet cependant pas de garantir ni le débit ni une latence de transmission puisque le mécanisme de souscription à l'interrogation régulière est réalisé en mode aléatoire et les interrogations, bien qu'elles restent égalitaires, peuvent être différées dans le temps.

Dans le but d'introduire une vraie QoS au niveau MAC, un groupe de discussion (TG11e) a été mis en place par l'IEEE. Ce groupe a proposé un amendement [IEE1 05] au standard de base IEEE 802.11 qui a été validé en juillet 2005. 802.11e prévoit deux mécanismes pour la gestion de la Qualité de Service dans 802.11 : *Enhanced DCF* et *Hybrid CF* [YU 04].

### 3.3.3.1 Enhanced DCF

*Enhanced DCF* (EDCF), parfois désigné par *Enhanced DCF Channel Access* (EDCA) propose une amélioration de l'original DCF en introduisant 8 classes de trafic. En interfaçant finement l'EDCF avec les couches supérieures, les trafics nécessitant des limites de latence et/ou de débits (vidéo, téléphonie) peuvent être rendus prioritaires. EDCF permet donc de réaliser une *différenciation de services* au niveau MAC. Pour la réaliser, le groupe de travail 802.11e propose d'une part d'intervenir sur les temps inter-trames et d'autre part sur plusieurs instances pour la gestion des fenêtres de contention (CW) :

- **Temps inter-trames** : SIFS, PIFS et DIFS sont conservés, mais une nouvelle durée inter-trame de valeur variable et proportionnelle à chaque classe et supérieure à DIFS, est introduite : AIFS pour *Arbitration Inter Frame Space*.
- **Fenêtres de contention** : nous avons vu plus haut que les fenêtres de contention aléatoires permettent de désynchroniser les accès au médium concurrents pour limiter le phénomène des collisions. Nous avons également vu que si une station ne remporte pas l'accès au médium, elle conserve sa valeur restant à courir de CW pour avoir plus de chance de le remporter à la tentative suivante. Avec le système de classe de trafics, 802.11e propose de gérer plusieurs instances de cette fenêtre de contention pour chaque trafic.

L'usage de ces deux nouvelles fonctionnalités permet de considérer jusqu'à 8 *stations virtuelles*, au sein d'une même station « physique » ; chaque station virtuelle possède sa propre FIFO pour les données, son AIFS et sa propre instance de CW et chaque station virtuelle est gérée indépendamment des autres.

De plus, chaque niveau de priorité est associé à une valeur désignée par *opportunité de transmission* (*Transmit Opportunity*, TXOP). TXOP définit un intervalle de temps pendant lequel une station peut conserver l'accès au médium et envoyer autant de trames qu'elle le souhaite. L'introduction de TXOP permet de mieux répartir les accès au médium ; dans le cadre de 802.11 DCF, les stations nécessitant un accès bas débit étaient favorisées par rapport à celles nécessitant un haut débit.

EDCF permet donc d'introduire des niveaux de priorités (différenciation de services) au niveau MAC [GARC 02], un peu comme le permettait la méthode d'accès au médium EY-NPMA utilisée dans HiperLAN/1. Il permet donc à certains trafics d'être plus prioritaires que d'autres [CHOI 03] ; en revanche, aucune garantie n'est possible et les trafics peuvent toujours entrer en collision.

### 3.3.3.2 Hybrid CF

*Hybrid CF* (HCF), parfois désigné par *Hybrid CF Channel Access* (HCCA) propose une gestion de l'accès au médium *orientée déterminisme* qui améliore le PCF initial de 802.11 en le rendant plus souple. Le problème majeur du PCF est que les dernières stations interrogées par le point d'accès pouvaient voir leur interrogation différée à la supertrame suivante. Avec 802.11 PCF, la limite entre zone sans

contention (CFP) et zone avec contention (CP) est fixée d'avance par le Point de Coordination (le point d'accès) et annoncée dans le *beacon*, alors que la durée d'interrogation des stations est variable ; de ce fait, si les premières stations interrogées utilisent copieusement le médium, il ne restera pas assez de temps pour les dernières et l'accès au médium leur sera différé.

Dans le cadre de 802.11e [IEE1 05], HCF cette limite entre CFP et CP est variable, le Point de Coordination pouvant décider de « rajouter du temps » à la CFP pour permettre l'interrogation de toutes les stations.

### 3.3.3.3 Autres améliorations

802.11e propose quelques améliorations supplémentaires évoquées ici à titre indicatif :

- *Automatic Power Save Delivery* (APSD) est une amélioration significative de la gestion énergétique dans 802.11. Il permet à un point d'accès de tamponner les données à destination d'un nœud endormi jusqu'à ce que ce nœud signale son réveil en transmettant une trame de données. Dès que le point d'accès reçoit la trame, il renvoie les données en attente. APSD est efficace si la bande passante utilisée est symétrique, ce qui est le cas des applications de téléphonie IP pour lesquelles il a été conçu.
- *Block Acknowledgments* (BA) permet au destinataire d'acquitter toutes les trames d'un TXOP par un seul acquittement. BA permet la réduction de l'*overhead*<sup>24</sup> provoqué par des acquittements multiples.
- *NoAck* permet la mise en place de classes de services avec lesquels les messages transmis ne sont pas acquittés. Cette amélioration permet d'éviter la retransmission inutile de données à haute criticité temporelle.
- *Direct Link Setup* (DLS), ou *Direct Link Protocol* (DLP) permet l'envoi direct de données de station à station d'un même BSS ; cette fonctionnalité permet en fait l'envoi de messages dans un pseudo mode ad hoc, c'est-à-dire sans passer par le point d'accès, ce qui permet de multiplier par près de deux la bande passante dans le cas d'un transfert de station à station.

### 3.3.4 Bluetooth

De part l'utilisation de l'étalement de spectre par sauts de fréquence (FHSS), Bluetooth impose un fonctionnement en maître/esclave au niveau des couches physique et liaison de données, comme cela a été évoqué dans le paragraphe 1.5.3. De ce fait, la méthode d'accès est naturellement proche de TDMA où le maître, élément central, distribue les temps de parole sur le médium. A ce titre, Bluetooth propose plusieurs « méthodes d'utilisation du médium » dont les deux principaux sont des liens ACL (asynchrones, orientés « données ») et SCO (synchrones, orientés « voix »).

- Les liens SCO proposent un débit fixe de 64 kbits/s avec un délai d'acheminement minimal grâce à une absence d'acquittement. En contrepartie, même si les liens SCO peuvent mettre en œuvre une certaine redondance des données par l'emploi de FEC (*Forward Error Correction*), les liens SCO ne sont pas *fiables* dans la mesure où, même si une trame est détectée comme erronée par le récepteur, elle ne fera pas l'objet d'une retransmission. Typiquement, les liens SCO sont dédiés au transport de la voix où la présence d'un échantillon erroné provoque une sensation moins désagréable à l'oreille qu'un retard.

---

<sup>24</sup>L'*overhead* est le volume de données utile (part du débit offert) perdu car utilisé dans les en-têtes et les messages de gestion du réseau (acquittements, etc.).

- En revanche, les liens ACL permettent la fiabilité mais sont sujets à une gigue de transmission importante, à cause des retransmissions. Les liens ACL sont typiquement dédiés au transport de données nécessitant une fiabilité certaine mais non contraintes temporellement.

De part leur structure intrinsèque, les liens SCO sont très adaptés aux transports à fortes contraintes temporelles. Cependant, les spécifications Bluetooth ne prévoyant aucune autre application que le transport de la voix pour les liens SCO, les données numériques empruntant des liens SCO sont codés selon des lois  $A$  ou  $\mu$  par des CODECS internes et sont difficilement exploitables pour d'autres applications. De plus, le débit figé de 64 kbits/s, qui plus est symétrique, restreint considérablement les applications potentielles. Des travaux ont néanmoins été entrepris pour utiliser les liens SCO pour le transport de données de capteurs [ELHO 04].

En revanche, on trouve dans la littérature [VAND 01] [MERC 03] des travaux évoquant des fonctionnalités de gestion de la Qualité de Service sur les liens ACL de Bluetooth. Dès la première version de la norme [IEEE1 02], des paramètres ont été prévus au niveau L2CAP<sup>◇</sup> :

- *Type de service* : *Best Effort* ou *Guaranteed*, active ou désactive la gestion de la QoS,
- *Token Rate* permet de spécifier le débit moyen (en octets/seconde),
- *Token Bucket Size* spécifie la taille maximale d'une raffale de données (en octets),
- *Peak Bandwidth* spécifie le débit maximum autorisé (en octets/seconde),
- *Latancy* spécifie le délai maximum tolérable pour l'application entre le moment où les données sont reçues par le modem et le début de la transmission effective sur radio (en millisecondes),
- *Delay variation* spécifie la différence entre le minimum et le maximum du délai tolérable par l'application.

D'après la présentation de ces paramètres proposés par le standard, Bluetooth apparaît comme la technologie WLAN/WPAN la plus aboutie en matière de gestion de la Qualité de Service au niveau de la méthode d'accès au médium. Malheureusement, encore aujourd'hui, il est impossible de trouver un module implémentant toutes ces fonctionnalités sur le marché. Il semblerait que, comme dans le cas de 802.11 avec le mode PCF, les industriels ne parviennent pas à trouver un marché grand-public pour ces protocoles proposant des fonctionnalités évoluées et se contentent de commercialiser des équipements ne proposant que les fonctionnalités basiques des normes et des protocoles.

## 4 Conclusion

Dans ce chapitre, nous avons pu constater qu'il existe un large choix en matière de technologies WLAN/WPAN. Les normes et les produits sont nombreux; certains présentent des fonctionnalités originales et intéressantes, d'autres proposent des services comparables et équivalents, mais tous permettent de rendre très abordable une technologie électronique de pointe (très hautes fréquences, micro-électronique complexe) par le grand public.

L'étude sur les protocoles de Qualité de Service de niveau *réseau*, très répandus dans le monde IP, nous a permis de conclure sur la réelle nécessité de penser la QoS au niveau 2 lorsque l'on envisage de déployer un protocole à QoS de niveau 3 et supérieurs. En effet, bien que le médium de transmission ne soit pas parfait et que, par conséquent, on constate toujours des erreurs au niveau de la couche *physique* ( $BER \neq 0$ ), il est absolument nécessaire de limiter les pertes d'informations dans les niveaux supérieurs lorsque les bits, au niveau *physique*, sont transmis correctement, *a fortiori* lorsque le médium présente une faible fiabilité comme en sans fil.

Nous avons vu qu'il existe de nombreuses techniques d'accès au médium mais que peu proposent des fonctionnalités pour éviter totalement le phénomène des collisions, première source de pertes de données au dessus du niveau *physique*. Si certaines méthodes comme le *TDMA Dynamique* d'HiperLAN/2, le PCF de 802.11 ou le HCF de 802.11e permettent de garantir un accès régulier au médium sans collision, aucune de ces technologies ne propose une méthode permettant un accès totalement déterministe au médium, avec des garanties sur le plan temporel. Dans le chapitre suivant, nous présenterons la technologie IEEE

802.15.4 qui propose des fonctionnalités intéressantes sur ce point. Même si la gestion de la QoS dans 802.15.4 n'est pas parfaite, les produits compatibles sont disponibles et permettent une reprogrammation partielle ou totale de la couche MAC, laissant entrevoir des possibilités de prototypage intéressantes.





# Bibliographie du chapitre 1

- [APOS 96] C. APOSTOLAS, R. TAFAZOLLI et B.G. EVANS – *Comparison between elimination yield non pre-emptive priority multiple access (EY-NPMA) and dynamic TDMA (D-TDMA)* – 7th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRCap96) vol 2, issue 15-18 Oct 1996 pp 663-667 (1996)
- [ARCE] Le site web de l'ARCEP (Autorité de Régulation des Communications Electroniques et des Postes) – <http://www.arcep.fr>
- [BADI 04] H. BADIS et K. AL AGHA – *QOLSR Multi-path Routing for Mobile Ad Hoc Networks Based on Multiple Metrics* – 59th IEEE Vehicular Technology Conference (IEEE VTC'04-Spring), Milan, Italie (Mai 2004)
- [BOUZ 03] A. BOUZOUALEGH, T. VAL, F. PEYRARD et E. CAMPO – *Study of the Characteristics Needed for Underwater Acoustic Networks* – 7th WSEAS International Conference on Circuits, Systems, Communications and Computers (CSCC'2003), Corfu, Grece (Juillet 2003)
- [CAPP 02] Communauté d'agglomération Pau Pyrénées – PAU BROADBAND COUNTRY (*large bande pour tous*) – Dossier de présentation (Février 2002)
- [CHIC 05] O. CHICHEPORTICHE – *Airbus passe à la VoIP avec France Télécom* – <http://www.silicon.fr/fr/silicon/news/2005/07/12/airbus-passe-voip-france-telecom>
- [CHOI 03] S. CHOI, J. DEL PRADO, S. SHANKAR et S. MANGOLD – *IEEE 802.11e Contention-Based Channel Access (EDCF) Performance Evaluation* – IEEE International Conference on Communications (ICC'03) vol.2, pp 1151-1156 (Mai 2003)
- [CULL 04] D. CULLER, D. ESTRIN et M. SRIVASTAVA – *Overview of Sensor Networks* – IEEE Computer vol.37, issue 8 (Août 2004)
- [DESG 03] G. DESGEORGES – *Réseaux haut débit et réseaux sans fil* – Cours d'ingénieur ESEO, spécialité R&T (2003)
- [ELHO 04] S. EL HOMSI, E. CAMPO, T. VAL et J.J. MERCIER – *Utilisation de la technologie Bluetooth pour la transmission de données numériques dans les applications à contraintes temporelles variées* – Congrès des doctorants de l'Ecole Doctorale de Génie Electrique, Electronique et Télécommunications (GEET), Supaero, Toulouse (Mars 2004)
- [ETS1 06] ETSI PROJECT BROADBAND RADIO ACCESS NETWORKS – *Broadband Radio Access Networks (BRAN) : HiperMAN Physical (PHY) Layer* – ETSI Technical Specification ETSI TS 102 177 V1.3.1 (Février 2006)
- [ETS2 06] ETSI PROJECT BROADBAND RADIO ACCESS NETWORKS – *Broadband Radio Access Networks (BRAN) : HiperMAN Data Link Control (DLC) Layer* – ETSI Technical Specification ETSI TS 102 178 V1.3.2 (Mars 2006)
- [ETSI 00] ETSI PROJECT BROADBAND RADIO ACCESS NETWORKS – *Broadband Radio Access Networks (BRAN) : HiperLAN Type 2 System Overview* – ETSI Technical report ETSI TR 101 683 V1.1.1 (Février 2000)
- [ETSI] Le site web de l'ETSI (*European Telecommunications Standards Institute*) – <http://www.etsi.org>
- [FON] Le mouvement FON – <http://www.fon.com>
-

- [GARC 02] J.A. GARCÍA-MACÍAS, F. ROUSSEAU, G. BERGER-SABBATEL, L. TOUMI et A. DUDA – *Différenciation des services sur les réseaux sans-fil 802.11* – 9<sup>ième</sup> Colloque Francophone sur l'Ingénierie des Protocoles (CFIP'2002), Montréal, Québec (Mai 2002)
- [GOLD 01] M. GOLDHAMMER – *ETSI BRAN HiperMAN Liaison Report 2001-11-08*
- [LANS 99] J. LANSFORD – *HomeRF : Bringing Wireless Connectivity Home* – Intel HomeRF technology Tutorial (Avril 1999)
- [HOWE 87] D.A. HOWE – *Ku-band Satellite Two-Way Timing Using a Very Small Aperture Terminal (VSAT)* – 41st Annual Symposium on Frequency Control pp 149-160 (1987)
- [HOYM 01] C. HOYMAN, M. PÜTTNER et I. FORKEL – *The HiperMAN Standard - a Performance Analysis*
- [IEE1 02] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.15.1 IEEE Standard for Information technology, Part 15.1 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs)* – IEEE Std 802.15.1-2002 (2002)
- [IEE1 03] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.15.3 IEEE Standard for Information technology, specific requirements part 15.3 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for high rate wireless personal area networks (WPANs)* – IEEE Std 802.15.3-2003 (2003)
- [IEE1 04] IEEE COMPUTER SOCIETY and IEEE MICROWAVE THEORY AND TECHNIQUES SOCIETY – *802.16 IEEE Standard for Local and metropolitan area networks – part 16 : Air Interface for Fixed Broadband Wireless Access Systems* – IEEE Std 802.16-2004 (Octobre 2004)
- [IEE1 05] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.11 IEEE Standard for Information technology, Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Amendment : Medium Access Control (MAC) Quality of Service (QoS) Enhancements* – IEEE (2005)
- [IEE1 99] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.11 IEEE Standard for Information technology, Specific Requirements – Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications* – IEEE standard 8802-11 (1999)
- [IEE2 03] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.15.4 IEEE Standard for Information technology, Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)* – IEEE Std 802.15.4-2003 (2003)
- [IEEE] Le site web du comité de standardisation IEEE 802 (*IEEE 802 LAN/MAN Standards Committee*) – <http://www.ieee802.org>
- [IRDA] Le site web de IrDA (*Infrared Data Association*) – <http://irda.org>
- [LEPA 91] F. LEPAGE – *Les Réseaux Locaux Industriels - Principes illustrés par des exemples* – Hermes (1991)
- [MERC 03] A. MERCIER, P. MINET et L. GEORGES – *Introducing QoS support in Bluetooth Piconet with a Class-Based EDF Scheduling* – Rapport de recherche de l'INRIA RR-5054 (Décembre 2003)
- [MICH 05] F. MICHAUT et F. LEPAGE – *Adaptation des applications distribuées à la qualité de service du réseau de communication* – *Ecole d'été temps réel 2005 (ETR'05)* Nancy, France (Septembre 2005)
- [MUHL 02] P. MUHLETHALER – *802.11 et les réseaux sans fil* – Eyrolles (2002)
- [NIC1 98] K. NICHOLS, S. BLAKE, F. BAKER et D. BLACK – *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers* – RFC 2474, IETF Network Working Group (Decembre 1998)
- [NIC2 98] K. NICHOLS, S. BLAKE, F. BAKER et D. BLACK – *An Architecture for Differentiated Service* – RFC 2475, IETF Network Working Group (Decembre 1998)
- [PUJO 95] G. PUJOLLE – *Les réseaux* – Eyrolles (1995)
- [RAN] Le web ring des RAN (*Rural Area Network*) – <http://ran.vaour.net/webring>

- [SIMP 05] D. SIMPLOT-RYL – *Real-time aspects in Wireless Sensor Networks* – Ecole d’Eté Temps Réel 2005 (ETR’05), Nancy, France (Septembre 2005)
- [SMER 04] S.A. SMERZI, G. GIRLANDO, T. COPANI et G. PALMISANO – *A Ku-band monolithic receiver for DVB-S applications* – IEEE Communications Magazine, vol.42 issue 8, pp 132-139 (Août 2004)
- [TSF] Le site web de l’association TOULOUSE SANS FIL – [http ://www.toulouse-sans-fil.net](http://www.toulouse-sans-fil.net)
- [VAND 01] M. VAN DER ZEE et G. HEIJENK – *Quality of Service in Bluetooth Networking* – 10/0362-FCP NB 102 88 Uen (Janvier 2001)
- [WEI 02] A. WEI et S. BOUMERDASSI – *Support de la QoS dans les réseaux 802.11* – Rapport scientifique CEDRIC N° 401 (2002)
- [WEIN 97] J. WEINMILLER, M. SCHLÄGER, A. FESTAG et A. WOLISZ – *Performance Study of Access Control in Wireless LANs - IEEE 802.11 DFWMAC and ETSI RES 10 HiperLAN – Mobile Networks and Applications*, vol.2, pp 55-67 (1997)
- [WROC 97] J. WROCLAWSKI – *The Use of RSVP with IETF Integrated Services* – RFC 2210, IETF Network Working Group (Septembre 1997)
- [XU 02] K. XU, M. GERLA et S. BAE – *How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks* – IEEE Global Telecommunications Conference (GLOBECOM’02) vol.1, pp 72-76 (Novembre 2002)
- [YU 04] J. YU – *IEEE 802.11e QoS for Wireless LAN : A Research Direction* – Multimedia Networking Research Laboratory (MNLAB) Networking Seminar (Décembre 2003)



## Chapitre 2

# Présentation des normes IEEE 802.15.4 / ZigBee

*Ce chapitre présente les normes et spécifications des technologies réseaux personnels sans fil IEEE 802.15.4 et ZigBee sur lesquelles sont basés les travaux effectués durant la thèse. Nous verrons dans ce chapitre que ces deux technologies sont intimement liées, mais aussi qu'elles présentent certaines failles que nous proposerons de résoudre par la suite, dans le chapitre 3.*

---

<b>1</b>	<b>Généralités</b>	<b>53</b>
1.1	Le projet ZigBee	53
1.2	Objectifs et domaine d'application	53
1.3	Consommation énergétique	53
1.4	Implémentations	54
1.5	Topologies	55
1.5.1	Topologie étoile	55
1.5.2	Topologie point à point	55
1.5.3	Topologies plus complexes	55
1.6	Adressage	56
1.7	Valeurs typiques	56
<b>2</b>	<b>Présentation de la pile protocolaire ZigBee</b>	<b>56</b>
2.1	Quelques notions fondamentales	56
2.1.1	Le découpage en différentes couches	57
2.1.2	Le principe de l'encapsulation	57
2.1.3	Protocole d'échange entre deux couches voisines	58
2.1.4	Représentation de la pile protocolaire ZigBee	59
2.1.5	Les interfaces de communication entre couches	59
2.2	La couche Physique	60
2.2.1	Bandes de fréquences et canaux	60
2.2.2	Modulations, étalement de spectre	61
2.2.3	Portée, puissance d'émission et sensibilité du récepteur	61
2.2.4	Le paquet de niveau physique	62
2.2.5	Services rendus	63
2.3	La couche Liaison	63
2.3.1	La sous-couche MAC	63
2.3.1.1	Types d'accès au médium	63
2.3.1.2	Notion de supertrame	65
2.3.2	La sous-couche LLC	66

2.3.3	Structures de trames . . . . .	67
2.3.3.1	Le champ de contrôle de trame . . . . .	67
2.3.3.2	Le champ de numéro de séquence . . . . .	68
2.3.3.3	Le champ d'adressage . . . . .	68
2.3.3.4	Le champ <i>payload</i> . . . . .	68
2.3.3.5	Le champ FCS . . . . .	69
2.4	La couche Réseau . . . . .	70
2.4.1	Éléments de la topologie du réseau . . . . .	70
2.4.1.1	Topologie en arbre . . . . .	70
2.4.1.2	Topologie maillée . . . . .	72
2.4.2	Architecture de la couche réseau . . . . .	72
2.4.3	Services rendus . . . . .	73
2.4.4	Adressage . . . . .	73
2.4.4.1	Adressage NHLE-based . . . . .	73
2.4.4.2	Adressage en arbre . . . . .	73
2.4.5	Principes de base du routage ZigBee . . . . .	74
2.4.5.1	Algorithme de routage à la demande . . . . .	75
2.4.5.2	Algorithme de routage hiérarchique : Tree Routing . . . . .	75
2.4.6	Ordonnancement des beacons dans un réseau avec arborescence . . . . .	76
2.4.7	Structure du paquet de niveau réseau . . . . .	76
<b>3</b>	<b>Identification des failles dans la méthode d'accès proposée par le standard</b>	<b>77</b>
3.1	La gestion des slots garantis ne présente pas une garantie absolue . . . . .	77
3.2	Autres problèmes identifiés . . . . .	78
3.3	Conclusion : les améliorations proposées . . . . .	79

---

# 1 Généralités

ZigBee est une technologie de réseau sans fil personnel (WPAN<sup>◇</sup>) destinée à l'électronique embarquée à très faible consommation énergétique. Elle propose une pile protocolaire propriétaire et légère, déclinable dans plusieurs versions plus ou moins complètes, pour des applications de transferts de données à faibles débits et de faibles taux d'utilisation du médium.

## 1.1 Le projet ZigBee

L'idée initiale du projet ZigBee date de 1998 ; une première proposition (v0.1) a été présentée courant 2000 puis rapidement une seconde (v0.2) à la fin de la même année. Après une soumission à l'IEEE mi 2001, la ZigBee Alliance [ZBAL] a été créée pour développer et promouvoir la norme. La production de modules compatibles fût alors prévue et les premiers produits (puces radio, piles protocolaires, modules intégrés, kits de développement, etc.) sont apparus et sont disponibles depuis la fin de l'année 2004.

ZigBee s'appuie sur la norme IEEE 802.15.4 [IEE2 03] [CALL 02] (que nous désignerons par la suite par « 802.15.4 ») pour les couches *physique* et *liaison*, qui sont les couches 1 et 2 du modèle OSI (cf. la figure 2.6). ZigBee propose ensuite ses propres couches supérieures (*réseau*, etc.) qui doivent faire l'objet d'une demande auprès de la ZigBee Alliance pour être utilisées. A ce titre, des droits sont perçus par la ZigBee Alliance pour tout emploi d'une pile ZigBee dans le cadre d'une application industrielle.

## 1.2 Objectifs et domaine d'application

ZigBee est un LP-WPAN<sup>◇</sup> : c'est un réseau sans fil à courte portée qui utilise les ondes hertziennes pour transporter des messages entre deux ou plusieurs entités réseaux. Il est caractérisé par une portée comprise entre quelques mètres et quelques centaines de mètres et un débit faible (max. 250kbits/s). La différence entre ZigBee et la plupart des autres WPAN existants se situe au niveau de l'utilisation du médium hertzien ; ZigBee est optimisé pour une faible utilisation du médium partagé par tous, par exemple 0.1% du temps [ZHEN 04]. Typiquement, un module ZigBee occupera le médium pendant quelques millisecondes en émission, attendra éventuellement une réponse ou un acquittement, puis se mettra en veille pendant une longue période avant l'émission suivante, qui aura lieu à un instant prédéterminé. Cette nécessité introduit des problématiques de recherche intéressantes, notamment au niveau des couches Liaison (temporisation et stockage des messages, accès original au médium) et Réseau (routage avec respect de contraintes énergétiques).

ZigBee est conçu pour interconnecter des unités embarquées autonomes comme des capteurs/actionneurs, à des unités de contrôle ou de commande. De telles entités embarquées pouvant dès lors être alimentées pendant plusieurs mois par des piles standards.

## 1.3 Consommation énergétique

Le point fort de ZigBee est sa très faible consommation énergétique, grâce à un mode de fonctionnement appelé *doze* ou *somnolence*. Ce mode permet à une entité communicante ZigBee de consommer très peu d'énergie (100 $\mu$ W) tout en permettant de passer en mode opérationnel en très peu de temps (300  $\mu$ s, cf. tableau 2.1), contrairement à d'autres WPAN comme Bluetooth par exemple [VDB1 05] [KHOU 05].

Bien que les progrès de l'électronique en terme de consommation énergétique soient sensibles ces derniers temps (possibilités accrues et rapides de mise en veille, baisse significative des courants de fuites dans le silicium, etc.), les bonnes performances de ZigBee sur ce plan sont essentiellement dues au mode *somnolence*, qui implique par conséquent une faible utilisation protocolaire du médium. Un

réseau ZigBee utilisé en continu, par exemple pour une application de type *streaming audio* consommera autant d'énergie que tout réseau sans fil classique, à puissance d'émission et débit équivalents.

A titre d'exemple, la consommation énergétique d'un modem ZigBee type fabriqué par FREESCALE [FRE1 04] est donnée dans le tableau 2.1. Le modem considéré est le MC13192 [FRE2 04] [FRE3 04]. Notons que le mode de fonctionnement normal du module est le mode repos, et de ce fait, les temps de basculement sont donnés par rapport à cet état de référence.

Mode	Consommation	Temps vers « repos »
Off		23ms
Hibernation	3 $\mu$ A	18ms
Somnolence	☺ 40 $\mu$ A ☺	☺ 332 $\mu$ s ☺
Repos	1mA	
Emission	34mA	144 $\mu$ s
Réception	37mA	144 $\mu$ s

TAB. 2.1 – Consommation typique d'un modem 802.15.4 avec temps de basculement entre chaque état (source FREESCALE)

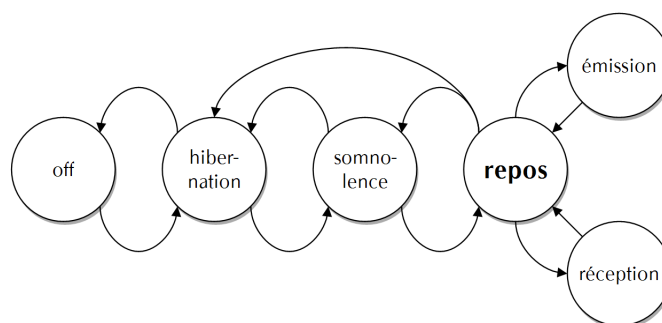


FIG. 2.1 – Graphe des états d'un modem 802.15.4 (source FREESCALE)

## 1.4 Implémentations

La technologie ZigBee prévoit deux types d'entités :

1. les entités complètes, ou FFD<sup>◇</sup>,
2. les entités réduites, ou RFD<sup>◇</sup>.

Les FFD implémentent la totalité de la spécification ZigBee alors que les RFD sont des entités allégées dans un objectif de moindre consommation énergétique et de moindre utilisation mémoire pour le micro-contrôleur. Les entités RFD sont nécessairement des nœuds terminaux du réseau ; un tel nœud ne saura pas router un paquet sur le réseau. Typiquement, un capteur embarqué sera RFD et alimenté sur batteries, alors qu'une unité centrale de traitement, alimentée par une source non contrainte énergétiquement « *main powered* », sera FFD avec une fonction de coordination du réseau, comme nous verrons dans le paragraphe suivant.



## 1.5 Topologies

Selon les besoins de l'application, la norme IEEE 802.15.4 prévoit deux topologies : étoile (*star*) ou point à point (*peer to peer*). Le réseau formé est appelé PAN<sup>◇</sup>. Ces deux topologies sont représentées en figures 2.2 et 2.3. Au dessus de 802.15.4, la couche réseau de ZigBee permet la création de réseaux plus complexes comme les réseaux maillés (*mesh*) ou arborescents (*tree*) grâce à un routage automatique des paquets de niveau 3 (niveau *réseau*).

### 1.5.1 Topologie étoile

Dans la topologie étoile, les entités RFD sont connectées à un nœud FFD central appelé *coordonateur*<sup>1</sup> ; dans cette topologie, tous les messages sont relayés par le coordonnateur, comme dans un *Piconet Bluetooth* avec le maître ou dans un réseau *WiFi en mode infrastructure* avec le point d'accès. Les communications directes entre entités RFD sont impossibles. Notons que le rôle central du coordonnateur implique de plus fortes dépenses énergétiques ; un coordonnateur devra donc généralement prévoir une source d'alimentation non contrainte (batteries conséquentes, secteur) [LLI2 06].

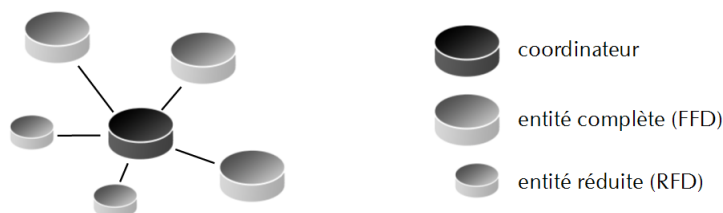


FIG. 2.2 – Représentation de la topologie en étoile

### 1.5.2 Topologie point à point

Dans la topologie point à point (*peer-to-peer*), un FFD peut communiquer directement avec tout autre FFD si ils sont à portée radio l'un de l'autre. Dans cette topologie, on retrouve un coordonnateur unique comme dans la topologie étoile. Son rôle est de tenir à jour une liste des participants au réseau et de distribuer les *adresses courtes* (cf. paragraphe 1.6 de ce chapitre).

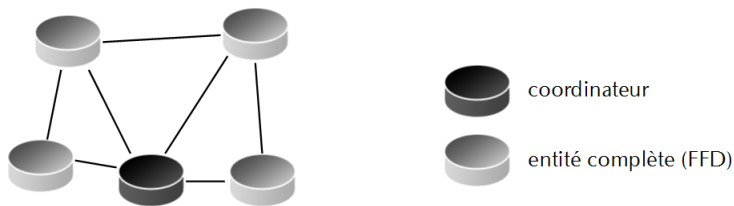


FIG. 2.3 – Représentation de la topologie point à point

### 1.5.3 Topologies plus complexes

Avec l'aide d'une couche réseau et d'un système de routage des paquets de données, il est possible d'élaborer des topologies plus complexes. La technologie ZigBee propose une couche réseau permettant de créer facilement de telles topologies grâce à des algorithmes de routage automatique tels que le *cluster*

<sup>1</sup>Dans la littérature, on retrouve également la dénomination *contrôleur primaire du réseau*.

*tree* (arborescence de cellules) ou les réseaux maillés *mesh*. Nous aborderons ces techniques plus loin dans cette thèse, dans la partie consacrée à la couche réseau (cf. paragraphe 2.4 de ce chapitre, page 70).

## 1.6 Adressage

Toute entité 802.15.4 possède une adresse unique appelée *adresse MAC*<sup>◇</sup>. A la différence de 802.3, une adresse MAC 802.15.4 a une longueur de 64 bits, soit 8 octets, contre 6 dans 802.3 ou 802.11. Dans 802.15.4, cette adresse est également appelée *adresse étendue*. Elle peut être utilisée dans les dialogues au sein du PAN, mais une seconde adresse appelée *adresse courte*, sur 16 bits, sera préférée dans la plupart des cas compte tenu des débits de transmission, relativement faibles. L'adresse courte est attribuée par le coordinateur du PAN au moment de l'*association au réseau*. La norme 802.15.4 ne prévoit pas de règle pour le choix de ces adresses, cette tâche est laissée au libre arbitre des couches supérieures. Nous verrons plus loin que dans la spécification de sa couche réseau, ZigBee propose un algorithme de distribution d'adresses automatique et décentralisé. Notons d'ores et déjà que ZigBee propose l'utilisation d'un adressage commun pour les couches 2 et 3, mais, à la différence d'autres protocoles comme IPv6, c'est la couche 3 qui impose son adresse à la couche 2.

## 1.7 Valeurs typiques

Pour conclure cette partie sur les généralités des deux normes, voici en résumé les valeurs typiques caractérisant IEEE 802.15.4 et ZigBee :

- Débit : 250 kbits/s sur le médium physique pour PHY2450,
- Puissance d'émission typique : entre 0 et 3 dBm,
- Portée radio : quelques centaines de mètre en espace libre,
- Consommation énergétique du composant d'émission / réception (hors traitement CPU) :
  - 3  $\mu$ A en hibernation (*hibernate mode*),
  - 40  $\mu$ A en somnolence (*doze mode*),
  - 1 mA au repos (*idle mode*),
  - 30 mA en émission,
  - 40 mA en réception.
- Taille de la pile protocolaire (code + mémoire) :
  - inférieure à 20 ko pour une entité réduite (RFD),
  - entre 40 à 60 ko pour une pile complète (FFD).
- Nombre d'entités connectables au réseau :
  - 256 dans une étoile ( $2^8$ ),
  - 65536 dans un PAN maillé ( $2^{16}$ ),
  - 18446744073709551616 adresses MAC disponibles ( $2^{64}$ )
- Accès au médium : pur CSMA/CA<sup>◇</sup> (sans RTS/CTS<sup>◇</sup>) ou organisé (mode balisé avec slots dédiés),
- Détection / correction d'erreurs : FCS<sup>◇</sup> 16 bits dans la trame MAC.

# 2 Présentation de la pile protocolaire ZigBee

## 2.1 Quelques notions fondamentales

Comme la plupart des technologies réseaux, l'ensemble des protocoles décrits par la norme ZigBee est représentable sous la forme d'une pile protocolaire découpée en plusieurs couches. Ce découpage permet de séparer clairement les différentes tâches ; l'identification des spécialités des différents acteurs et des différents métiers de la conception réseau est ainsi rendue plus claire.

### 2.1.1 Le découpage en différentes couches

La pile proposée par l'IEEE et la ZigBee Alliance suit les recommandations de l'ISO en terme de séparation des rôles attribués aux différentes couches. Comme cela a été dit précédemment, cette pile reprend les couches 1 et 2 normalisées dans la norme 802.15.4 et ajoute ses propres couches supérieures. La couche 1 (couche physique) décrit les caractéristiques de l'interface radio (fréquences, largeur de bande, modulation, débit binaire, etc.) ; la couche 2 (couche liaison) décrit les caractéristiques de la sous-couche MAC (gestion des accès au médium) et la sous-couche SSCS<sup>◇</sup> (formation de trame, convergence des données) ; la couche 3 (couche réseau) décrit le processus de routage des données sur le réseau ; enfin, la couche la plus haute (couche application) décrit le système élaboré de profils, à l'instar de Bluetooth ou IrDA (comme vu dans le chapitre 1), qui permet la normalisation du niveau *application* directement dans la pile protocolaire, au même titre que les couches basses.

*Une description plus précise du rôle et des caractéristiques de chaque couche est faite dans les sections suivantes.*

### 2.1.2 Le principe de l'encapsulation

Rappelons tout d'abord le principe générique de l'encapsulation : lorsque deux entités d'une même couche s'échangent des messages, les données sont de fait véhiculées par les couches inférieures. Les couches les plus hautes auront généralement une bonne connaissance de la raison de la communication, mais une piètre idée de ce qui se passe réellement sur le médium (le chemin emprunté par l'information, etc.). A l'opposé, les couches les plus basses auront une connaissance très précise sur la manière dont les données sont gérées et transportées, mais très peu d'informations sur le contenu réel et ce qu'elles représentent. Cependant, c'est le système dans son ensemble qui va permettre de transmettre des informations de manière fiable sur un réseau étendu et hétérogène.

Le rôle d'un réseau est bel et bien de transmettre des données ; les données que reçoit une couche N par la couche N+1 sont appelées les *données de service*. On parlera d'une *unité de données de service*, ou SDU<sup>◇</sup>, pour désigner un ensemble de données passées par la couche supérieure. Lorsqu'une couche veut transmettre un message, elle le transmet à la couche immédiatement inférieure en y ajoutant des informations que son homologue, l'entité de même couche à l'autre bout du réseau, saura comprendre pour traiter les données reçues. Ces informations ajoutées peuvent précéder les données à transmettre – on parlera alors d'*en-tête*, ou *header*, HR – ou bien les succéder – on parlera alors de *postambule*. L'ensemble passé à la couche inférieure (en-tête, données de service, postambule) formera l'*unité de données du protocole*, ou PDU<sup>◇</sup>. La couche inférieure traitera le PDU de la couche supérieure comme son SDU. Généralement, on ajoute la première lettre du nom de niveau pour désigner un HR, un PDU ou un SDU. Par exemple, un PDU de niveau MAC sera désigné par MPDU (MacPDU) et un SDU de niveau Physique sera désigné par PSDU (PhySDU). Notons qu'un MPDU est équivalent au PSDU, puisque les niveaux PHY et MAC sont mitoyens dans une pile protocolaire classique comme celle de ZigBee.

Le principe générique de l'encapsulation est illustré par la figure 2.4. Un exemple appliqué à 802.15.4 sera illustré plus loin dans ce chapitre par la figure 2.20, page 76.

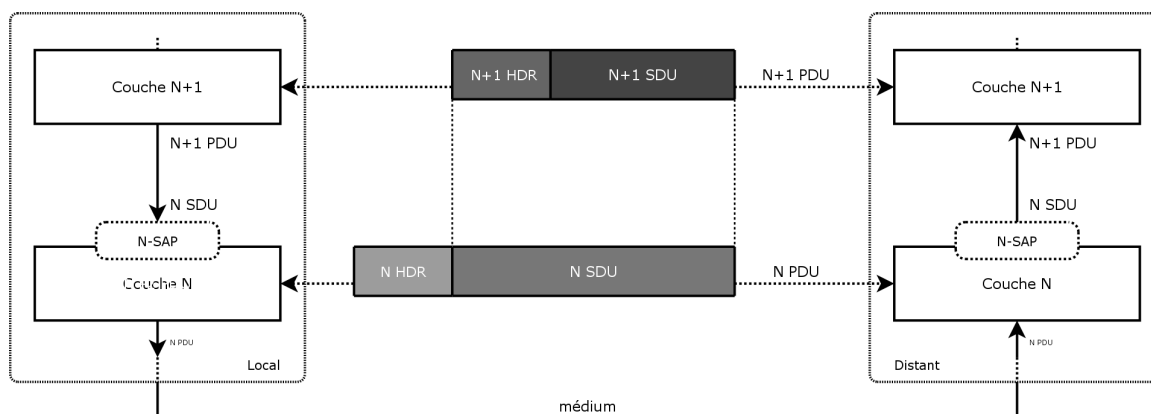


FIG. 2.4 – Le principe générique de l'encapsulation

### 2.1.3 Protocole d'échange entre deux couches voisines

Les échanges de messages entre deux couches mitoyennes se font selon le principe classique du *service de niveau  $n$* , service assuré par la  $n^{ième}$  couche de la pile. Pour accéder à ce service, la couche immédiatement supérieure ( $n + 1$ ) peut accéder au *Point d'Accès de Service* (SAP<sup>◇</sup>) de niveau  $n$ .

Pour être accessible, un SAP propose un jeu de *primitives* propre à ses capacités (par exemple : scanner le médium radio 2,4GHz, créer une connexion au réseau, envoyer des données, etc.). Selon ses besoins, la couche supérieure appelle les primitives qu'elle souhaite dans l'ordre imposé par son protocole.

Pour s'échanger des messages, deux couches mitoyennes disposent de quatre types de primitives :

1. *Request* : demande effectuée par le service local de niveau  $n + 1$  au SAP de niveau  $n$ .
2. *Indication* : la demande qui vient d'être faite par le *request* est acheminée jusqu'au nœud destinataire par le réseau ; le SAP distant de niveau  $n$  le transmet à la couche  $n + 1$ .
3. *Response* : la couche distante de niveau  $n + 1$  répond en envoyant un message à son entité paire. Pour être acheminé par le réseau, le message est passé au SAP de niveau  $n$ .
4. *Confirm* : le SAP local de niveau  $n$  transmet cette réponse au service de niveau  $n + 1$ , qui avait initié la requête.

La figure 2.5 illustre ce fonctionnement.

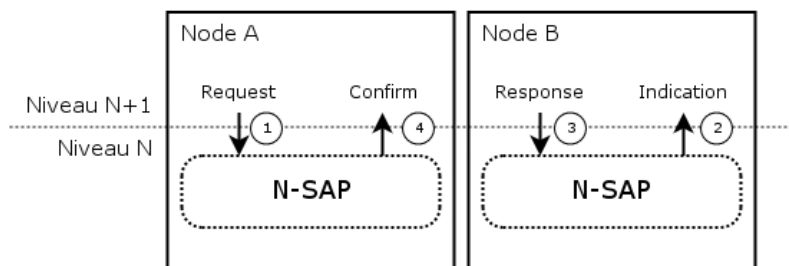


FIG. 2.5 – Communication respectant le protocole normalisé

### 2.1.4 Représentation de la pile protocolaire ZigBee

La pile protocolaire ZigBee est représentée sur la figure 2.6. On y retrouve le découpage classique recommandé par l'OSI. Une description fine des trois couches les plus basses – sur lesquelles portent le travail effectué pendant la thèse – sera faite dans la suite de ce chapitre.

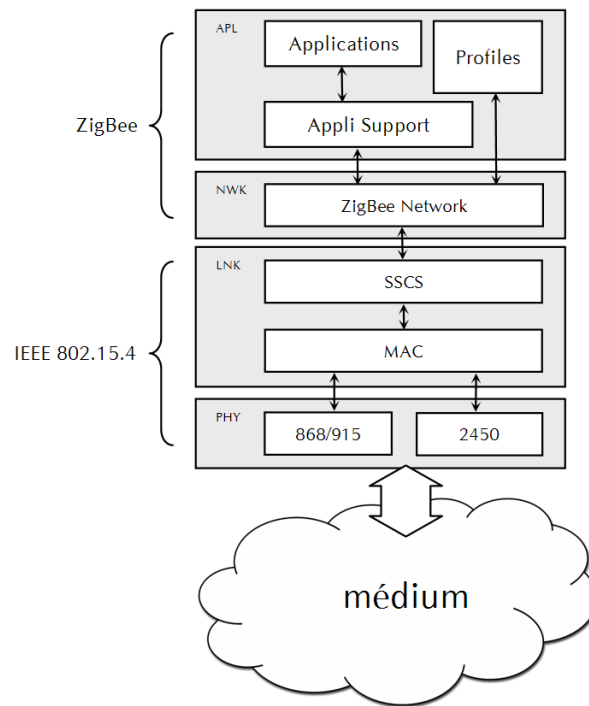


FIG. 2.6 – La pile protocolaire 802.15.4 / ZigBee

### 2.1.5 Les interfaces de communication entre couches

La norme ZigBee prévoit plusieurs interfaces pour la communication entre couches. Ces interfaces sont les Points d'Accès de Service (SAP) vus précédemment. La figure 2.7 représente cet interfaçage protocolaire.

A l'instar des autres piles protocolaires comme 802.11 [IEE1 99], chaque couche de la pile protocolaire ZigBee compte deux entités, chacune ayant sa propre interface :

- Une entité dédiée aux **transferts de données**. Cette entité est sollicitée lorsque la couche supérieure veut envoyer ou recevoir des données sur le réseau. Les primitives présentes sur cette interface sont généralement peu nombreuses (envoyer des données, recevoir des données) et elles se retrouvent à chaque niveau de la pile, pour chaque couche.
- Une entité dédiée à la **gestion de la couche**. Cette entité sert à commander toutes les tâches propres à la couche. Elle dispose généralement d'un jeu de primitives plus étoffé que l'interface dédiée aux transferts de données. Chaque couche ayant un rôle bien précis, chaque niveau dispose d'un jeu qui lui est propre. Voici quelques exemples de primitives :
  - pour le niveau Physique :
    - changer de canal radio,
    - passer en émission,

- détecter l'énergie sur le médium.
- pour le niveau Liaison :
  - rechercher un coordinateur,
  - se synchroniser sur une autre entité,
  - notifier que l'on quitte le réseau,
  - demander de la bande passante à un coordinateur (plus de droit à la parole).
- pour le niveau Réseau :
  - diffuser les routes connues,
  - demander une route.

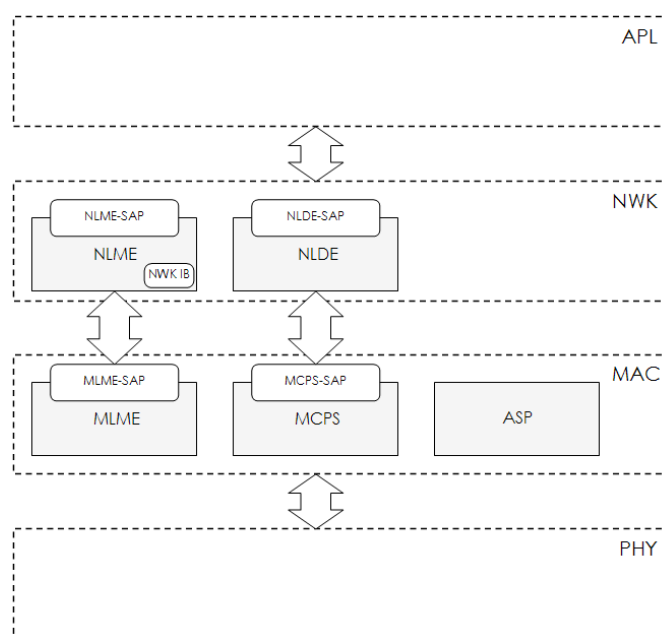


FIG. 2.7 – Principe d'interfaçage entre couches et SAP pour ZigBee

Notons que les constructeurs ajoutent parfois des entités et/ou des interfaces propres à leur produits, qu'ils soient matériels ou logiciels. C'est le cas du constructeur FREESCALE qui ajoute une interface ASP<sup>◇</sup> au niveau MAC de sa solution ZigBee [FRE4 04]. Cette interface permet de piloter les fonctions d'économie d'énergie du jeu de composants d'émission/réception. Bien entendu, la partie logicielle gérant le composant (*driver*) devra utiliser cette interface pour optimiser la solution réseau en terme de consommation énergétique.

## 2.2 La couche Physique

### 2.2.1 Bandes de fréquences et canaux

Conformément à IEEE 802.15.4, ZigBee peut travailler sur trois bandes de fréquences différentes : 868 MHz pour la région Europe, 915 MHz pour l'Amérique du nord, et 2,4 GHz pour une couverture mondiale. La norme prévoit deux couches physiques différentes (PHY), une pour le 868/915MHz (PHY868/915) et une seconde pour le 2,4 GHz (PHY2450). Le tableau 2.2 résume les caractéristiques et les paramètres des deux couches physiques proposées (PHY868/915 et PHY2450).

PHY	Bande (MHz)	Nbre canaux (n°)	Région	Étalement de spectre		Données		
				Débit Chip (kChip/s)	Modulation	Débit binaire (kbit/s)	Débit Symboles (kSymb/s)	Symboles
<b>868/915</b>	868 ~ 868.6	1 (0)	Europe	300	BPSK	20	20	Binares
	902 ~ 928	10 (1~10)	USA	600	BPSK	40	40	Binares
<b>2450</b>	2400 ~ 2483.50	16 (11~26)	Toutes	2000	O-QPSK	250	62.5	16-ary orthogonal

TAB. 2.2 – Caractéristiques des deux couches physiques proposées (PHY868/915 et PHY2450)

Au total, 27 canaux (numérotés de 0 à 26) sont répartis sur ces trois bandes, cette diversité en terme d'utilisation du spectre radio-fréquence permettant à la technologie de répondre aux nombreuses réglementations et d'être utilisable sur toutes les régions du globe. Soit  $k$  le numéro de canal compris entre 0 et 26, on peut alors obtenir la fréquence de chaque canal ainsi :

- $F_c = 868,3$  MHz, pour  $k = 0$
- $F_c = 906 + 2(k-1)$  MHz, pour  $k = 1, 2, \dots, 10$
- $F_c = 2405 + 5(k-11)$  MHz, pour  $k = 11, 12, \dots, 26$ .

### 2.2.2 Modulations, étalement de spectre

Comme beaucoup d'autres technologies WLAN/WPAN, 802.15.4 met en œuvre une modulation à spectre étalé. Une modulation utilisant *l'étalement de spectre* permet d'améliorer l'immunité aux interférences et aux multi-trajets, en sacrifiant quelque peu les performances en terme de débit. De plus, grâce au codage réalisé par la séquence pseudo-aléatoire, aussi appelée *PN-codes*<sup>◇</sup> qui permet de réaliser l'étalement [GUIL 05], la confidentialité des échanges est améliorée. Cependant, cette technique ne garantit pas les propriétés de *confidentialité* et d'*authentification* ; une partie de la *couche liaison* effectuera cette tâche.

La couche PHY868/915 est relativement simple et basique : les symboles sont binaires, grâce à l'emploi d'une modulation BPSK<sup>◇</sup> et un débit peu élevé (20 kbits/s pour le 868 MHz, 40 kbits/s pour le 915 MHz). En revanche, la couche PHY2450 propose une modulation plus complexe, O-QPSK<sup>◇</sup>, qui permet un débit plus intéressant.

### 2.2.3 Portée, puissance d'émission et sensibilité du récepteur

IEEE 802.15.4 prévoit une portée classique de quelques dizaines à quelques centaines de mètres suivant l'environnement considéré. La puissance maximale émise par un module 802.15.4 ou ZigBee n'est pas définie par la norme ; celle-ci est laissée d'une part à l'appréciation de l'autorité de régulation de la zone où est effectuée la transmission, et d'autre part au constructeur pour des questions évidentes d'autonomie énergétique du système dans lequel il est implanté. Néanmoins, la puissance typique recommandée est de 1 mW (soit 0 dBm) et la sensibilité du récepteur doit être meilleure que -85 dBm (à 2,4 GHz, pour un taux d'erreur paquet meilleur que 1%).

En pratique, un nœud ZigBee a une portée de quelques dizaines de mètres, jusqu'à une centaine de mètres en extérieur et sans obstacle. Notons que de part la robustesse de la couche physique, les portées d'un transceiver 802.15.4 sont comparables à celle d'un transceiver 802.11, mais avec une puissance d'émission plus faible : à rapport signal sur bruit (SNR<sup>◇</sup>) égal, 802.15.4 dispose d'un taux d'erreur bit

(BER<sup>◇</sup>) meilleur que les autres technologies sans fil proposées par l'IEEE, comme l'illustre le graphe de la figure 2.8.

Notons que comme pour toutes les technologies de réseau sans fil, la portée effective d'un transceiver 802.15.4 est très liée à sa puissance d'émission. Certains modules sont dotés d'un étage amplificateur en sortie HF et/ou d'un amplificateur à faible bruit en entrée, ce qui permet d'étendre considérablement la portée radio. A titre d'exemple, les modules XBEEPRO fabriqués par la société MAXSTREAM [MAXS] sont vendus pour une portée de 1,6 km en ligne de vue.

Notons que l'étendue d'un réseau ZigBee peut être largement supérieure à la portée effective de deux modules grâce à la topologie maillée (multi-sauts), comme nous le verrons plus loin dans ce chapitre. Enfin, comme c'est le cas dans tous les réseaux hertziens, la portée est largement conditionnée par la présence d'obstacles ou par la pollution du spectre radio-fréquence. A ce titre, le coordinateur du réseau peut décider d'effectuer un changement de canal à titre exceptionnel si le taux d'erreur sur le réseau est trop important.

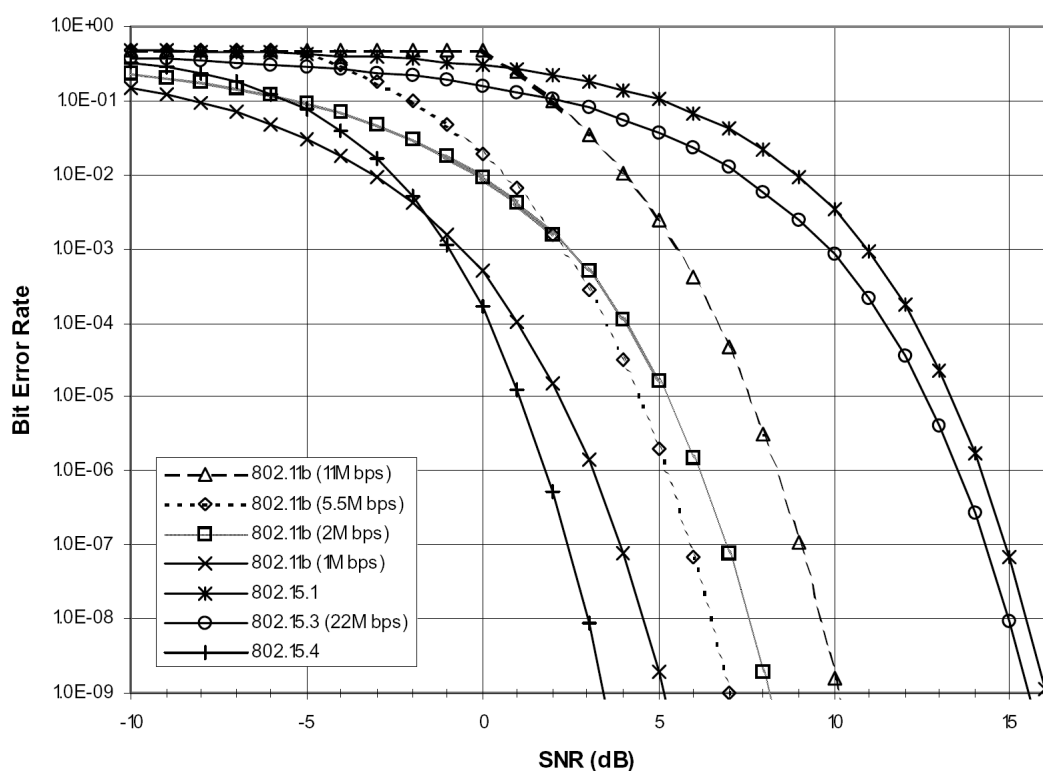


FIG. 2.8 – Rapport entre BER et SNR pour diverses technologies de transmission sans fil (source IEEE)

## 2.2.4 Le paquet de niveau physique

La norme prévoit un paquet<sup>2</sup> de niveau physique représenté sur la figure 2.9. Ce paquet comprend un en-tête de synchronisation, un en-tête PHY et les données PHY.

L'en-tête de synchronisation comprend 6 octets : un préambule d'une longueur de 5 octets qui permet au récepteur de parfaire sa synchronisation et un fanion de START sur 1 octet. Ce fanion rompt l'alternance des bits transmis lors du préambule, indiquant ainsi l'imminence de la transmission de données. Après l'en-tête de synchronisation vient l'en-tête PHY dont le rôle est de spécifier la longueur du

<sup>2</sup>Bien que le terme *paquet* soit généralement réservé au niveau 3 (réseau), c'est bien ce terme qui est utilisé par l'IEEE pour désigner ce groupe de bits envoyé sur le médium.



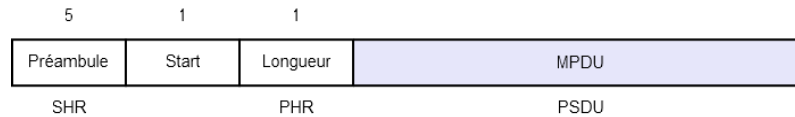


FIG. 2.9 – Structure du paquet de niveau physique

paquet. Les données suivent cet en-tête, 127 octets maximum par paquet, soit une durée de transmission maximum de 4,256 ms par paquet sur la couche PHY2450 (250 kbits/s).

### 2.2.5 Services rendus

La couche PHY de 802.15.4 rend deux services :

- un service de données PHY (*PHY data service*), qui permet l'émission et la réception de PPDU<sup>◇</sup> sur le médium radio,
- un service de gestion PHY (*PHY management service*), qui permet l'interfaçage entre le logiciel et l'entité de gestion de la couche physique. Les capacités de cette entité sont les suivantes (liste exhaustive) : activation/désactivation du *transceiver*, ED<sup>◇</sup>, LQI<sup>◇</sup> sur les paquets reçus, sélection du canal, CCA<sup>◇</sup> pour le CSMA/CA de la sous-couche MAC.

## 2.3 La couche Liaison

De façon très similaire au modèle défini par le groupe 802 de l'IEEE, le niveau liaison de 802.15.4 (Niveau 2 OSI) comprend une sous-couche d'accès au médium (MAC) et une sous-couche de convergence (SSCS).

### 2.3.1 La sous-couche MAC

#### 2.3.1.1 Types d'accès au médium

La sous-couche MAC gère les accès au médium radio, résolvant notamment les problèmes d'accès concurrents. 802.15.4 propose deux modes pour l'accès au médium : un mode *non coordonné* (totalement CSMA/CA, sans RTS/CTS<sup>3</sup>) et un mode *coordonné*, ou *beacon mode*, disponible uniquement dans une topologie étoile où le coordinateur de cette étoile envoie périodiquement des trames balises (*beacon*) pour synchroniser les nœuds du réseau. L'emploi du mécanisme CSMA/CA dans le mode non coordonné est relativement classique et 802.15.4 n'offre que peu d'innovations par rapport aux autres technologies sans fil dans ce mode ; en revanche, le mode coordonné permet d'entrevoir des applications intéressantes mettant en œuvre une *Qualité de Service*.

- **Le mode non coordonné**, totalement CSMA/CA

Dans le mode non coordonné, il n'y a pas d'émission de beacon donc pas de synchronisation entre les différents nœuds du réseau. Les nœuds voulant émettre des données doivent utiliser le protocole CSMA/CA « non slotté », c'est-à-dire que le début d'une émission se fait dès que le médium est jugé libre, sans attendre le début d'un éventuel slot. Cependant, même si l'algorithme est dit « non slotté », il se base tout de même sur une unité temporelle discrète appelée *période de backoff* pour

<sup>3</sup>Se reporter au premier chapitre pour plus de détail sur cette technique

pouvoir retarder plus ou moins l'émission d'une trame et éviter les collisions.

L'implémentation du CSMA/CA dans 802.15.4 est très proche de celle de 802.11, à deux exceptions près :

1. Absence de RTS/CTS : à la différence de 802.11, la méthode d'accès avec contention de 802.15.4 ne prévoit pas de mécanisme pour réserver le médium avant de commencer à émettre les données. Ceci s'explique par le fait que dans un réseau 802.15.4, les trames DATA sont très courtes et le débit très faible et, de fait, l'utilisation d'un protocole de type RTS/CTS serait très coûteuse. Après avoir attendu un nombre aléatoire de périodes de *backoff*, si le médium est libre, la trame de données est émise.
2. Gestion des temps inter-trames : 802.15.4 en prévoit trois, du plus court au plus long :
  - $t_{ACK}$  est le délai imposé entre une trame de données et son acquittement. Il doit être le plus court possible, mais dans tous les cas supérieur à  $aTurnaroundTime$  (192  $\mu s$ ),
  - SIFS $^\diamond$ , pour *Short Inter-Frame Spacing*, suit les trames dites « courtes », c'est-à-dire d'une longueur inférieure ou égale à  $aMaxSIFSFrameSize$  (18 octets),
  - LIFS $^\diamond$ , pour *Long Inter-Frame Spacing*, suit les trames dites « longues », c'est-à-dire d'une longueur strictement supérieure à  $aMaxSIFSFrameSize$  (18 octets).

Dans un objectif d'économie d'énergie, cette gestion des temps inter-trames, en fonction de la longueur de la trame, est nécessaire compte tenu des faibles ressources de traitement (CPU) des nœuds récepteurs : plus le volume des données est important, plus le temps de traitement nécessaire est grand.

#### • Le mode coordonné, ou *balisé*

Dans le mode coordonné, une ou plusieurs entité(s) du réseau diffuse(nt) périodiquement des trames appelées *balises*, ou *beacon*. Tout membre du réseau qui entend cette balise peut utiliser la réception de cette trame pour se synchroniser avec son émetteur et se servir de lui comme relais. Ce mode de fonctionnement permet les meilleures performances sur le plan énergétique car une fois l'information transmise au relais, le nœud communicant peut *somnoler* ; de plus, les messages en attente étant stockés dans la mémoire du relais, un nœud peut choisir de se réveiller selon ses besoins, et demander alors les données en attente. On parle alors de *transfert de données indirect* dans une *topologie en étoile*, car tout échange sur le réseau passe par le relais. On appellera par la suite ce relais le *coordinateur d'étoile*. La figure 2.10 illustre le fonctionnement de transfert de données dans une étoile.

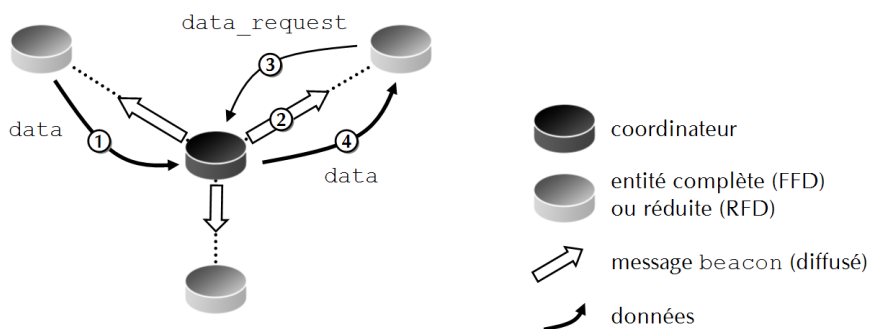


FIG. 2.10 – Principe du transfert de données dans une étoile

- Le transfert de données direct est illustré par l'envoi du message (1) de type *data*. Le nœud envoie directement ses données au coordinateur de l'étoile puis se rendort.

- Le nœud de destination récupère ses données de manière indirecte : le message **beacon** (2) annonce les données en attente pour tous les nœuds ; le nœud de destination écoute le beacon, constate que des données sont en attente et les réclame par le message **data\_request** (3). Le coordinateur peut alors transmettre les données en attente en envoyant le message **data** (4).

### 2.3.1.2 Notion de supertrame

L'espace temporel durant lequel les communications prennent part est appelé *supertrame*. Une supertrame est toujours divisée en 16 slots temporels de durées égales, la trame balise occupant le premier slot ; cette trame permet de diffuser la synchronisation pour tous les nœuds à portée radio, mais également l'identifiant du PAN et la structure dynamique de la présente supertrame, en fonction des demandes qui ont été faites par les nœuds membres de l'étoile. La structure d'une supertrame est représentée figure 2.11.

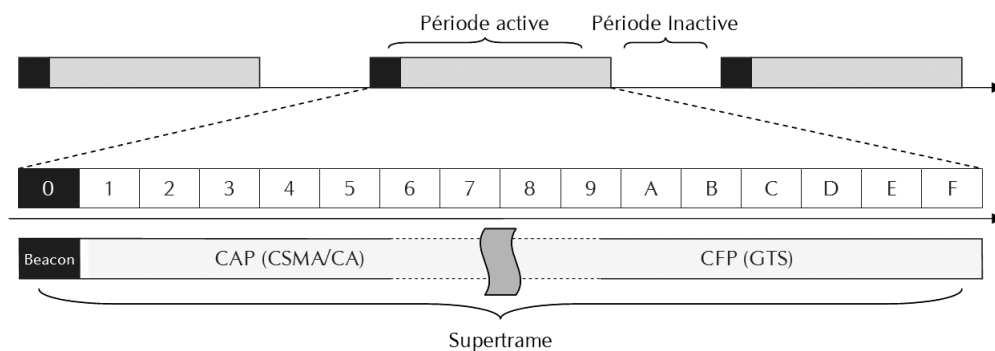


FIG. 2.11 – Représentation d'une supertrame IEEE 802.15.4

La supertrame possède deux paramètres fondamentaux :

- $BO^\diamond$  (*Beacon Order*), qui fixe l'intervalle de temps entre l'envoi de deux messages beacon par le coordinateur. On déduit de  $BO$  le paramètre  $BI^\diamond$  (*Beacon Interval*), selon la loi :

$$BI = 2^{BO} * 15,36 \text{ ms avec } 0 \leq BO \leq 14$$

- $SO^\diamond$  (*Superframe Order*). Ce paramètre fixe la durée active de la supertrame selon la loi :

$$d_{active} = 2^{SO} * 15,36 \text{ ms avec } 0 \leq SO \leq BO$$

L'organisation de l'accès au médium par supertrame permet les meilleures économies sur le plan énergétique. Les nœuds du réseau se réveillent juste avant le slot 0 et se mettent à l'écoute. A la réception du beacon, ils prennent connaissance de la structure de la supertrame qui débute : valeurs de  $BO$  et  $SO$ , présence de données en attente, etc. S'ils n'ont de données ni à émettre, ni à recevoir, ils peuvent somnoler jusqu'au beacon suivant.

Notons que plus  $BO$  et  $SO$  sont faibles, plus la fréquence des supertrames est élevée, donc plus le réseau est réactif (latence faible). En revanche, plus la différence entre  $BO$  et  $SO$  est grande, meilleures sont les économies sur le plan énergétique. Il faudra donc trouver un compromis entre ces deux constantes selon l'application.

Le mode coordonné de 802.15.4 propose deux méthodes d'accès au sein de la supertrame :

- **un mode avec contention** avec lequel les accès se font de façon « classique » selon le protocole CSMA/CA slotté. Cependant, les trames balises sont toujours émises périodiquement en début de supertrame, pour assurer la synchronisation entre les entités du réseau. Ce mode d'accès au

médium est toujours possible et une partie de la supertrame doit être systématiquement dédiée par le coordinateur aux accès se faisant dans ce mode. Cette partie de la supertrame est appelée CAP<sup>◇</sup> (cf. 2.11).

- **un mode sans contention**, « *contention free* », optionnel, avec lequel les accès au médium sont maîtrisés par le coordinateur. Ce mode peut être utilisé par les nœuds qui en font la demande. Si la capacité du réseau le permet, le coordinateur pourra choisir d'honorer la demande du nœud en lui allouant un ou plusieurs slots, dans la direction (émission ou réception) qui a été demandée ; on parle de GTS<sup>◇</sup>. Les GTS sont toujours placés en fin de supertrame, dans les derniers slots. On appelle cette partie de la supertrame CFP<sup>◇</sup>. Dans cette période, l'accès au médium est donc organisé et réparti par le coordinateur et les collisions sont rendues plus rares.

Ce mode sans contention rend possible une réservation de bande passante et peut offrir une certaine garantie sur le plan temporel. Pour se faire, le coordinateur peut attribuer jusqu'à 7 GTS, alors qu'un GTS peut occuper un ou plusieurs des 16 slots que comporte la supertrame ; le début de la supertrame, via la CAP, reste toujours en accès libre par CSMA/CA pour permettre l'accès aux transports ne nécessitant pas ou peu de garantie, de ne pas être bloqués par un nombre trop important de GTS. De fait, 802.15.4 prévoit une taille limite pour la CAP dans la supertrame (*aMinCAPlength* de 440 symboles, soit 7,04 ms et ce, quelques soient les valeurs de *BO* et *SO*). Notons dès maintenant, bien que cela soit longuement évoqué en fin de chapitre, que les *demandes de GTS* ainsi que les *demandes d'association* au réseau ne peuvent se faire que dans la CAP. Il est donc primordial, pour ne pas bloquer le réseau, de limiter la taille de la CFP au sein la supertrame.

### 2.3.2 La sous-couche LLC

Conformément à la plupart des standards 802.*n*, IEEE 802.15.4 propose une sous-couche de convergence de type LLC pour normaliser l'interfaçage des couches décrites par le standard avec une couche de niveau supérieur, typiquement une couche de niveau 3 compatible LLC. Cette convergence est assurée par la sous-couche SSCS qui est décrite par le standard.

Une couche de convergence typique doit jouer plusieurs rôles :

1. La vérification de l'intégrité des données reçues avant la remise à la couche supérieure, par exemple par utilisation conjointe d'un Code de Redondance Cyclique (CRC) et d'un mécanisme d'acquiescement,
2. Le contrôle de flux, afin d'éviter la saturation des tampons de réception et la perte éventuelle de données ou les débordements de mémoire,
3. La convergence d'adressage, c'est-à-dire la correspondance des adresses de niveau 3 (niveau *réseau*, l'adressage est généralement globalisé sur tout le réseau) avec les adresses de niveau 2 (niveau *liaison*, l'adressage est généralement local et limité au médium utilisé). La convergence d'adressage permet également la gestion des procédés de diffusion (*broadcast*) et de diffusion par classes (*multicast*).

Dans le cadre de 802.15.4, la sous-couche de convergence proposée est très simple, et ce pour plusieurs raisons :

1. Tout d'abord, et c'est le cas avec la plupart des technologies sans fil, dans IEEE 802.15.4, le mécanisme d'acquiescement sert non seulement à assurer à un émetteur que le destinataire a reçu correctement les données envoyées, mais aussi à détecter que la transmission sur le médium partagé n'a pas subi de collision (CSMA/CA). De ce fait, l'acquiescement est pris en charge par le processus de gestion d'accès au médium et le standard ne prévoit pas de le faire remonter jusqu'à la sous-couche SSCS ; la gestion de ce dernier reste cantonnée à la sous-couche MAC. De plus, la

présence d'un système évolué de gestion de la couche MAC nécessite l'intégrité des messages de gestion à ce niveau : cette vérification est donc assurée par un processus qui se situe en dessous de la couche liaison. La sous-couche de convergence n'a donc pas à assurer cette tâche une seconde fois.

2. Dans IEEE 802.15.4, le contrôle de flux est implicite et réalisé très simplement car, étant donné le faible volume de données échangées et la faible capacité mémoire des nœuds, le système d'envoi des paquets de données est de type *send & wait*, c'est-à-dire que la couche supérieure attend le traitement intégral du paquet de données (y compris la gestion de l'acquittement) avant l'envoi du paquet suivant (pas d'anticipation).
3. Dans IEEE 802.15.4, la gestion d'adressage est simplifiée dans la mesure où il n'y a pas de conversion à réaliser. Le plan d'adressage est le même entre la couche liaison et la couche réseau, les nœuds utilisent leur adresse MAC pour s'associer au réseau puis se voient attribués une *adresse courte* par le coordinateur du PAN.

La sous-couche SSCS propose un jeu de trois primitives :

1. **MA-UNITDATA.request** permet à une instance LLC d'ordonner l'envoi d'un paquet de données. Seuls trois paramètres sont alors passés : adresse source, adresse destination et données.
2. **MA-UNITDATA.indication** permet à l'entité 802.15.4 d'indiquer à l'entité LLC supérieure que des données valides ont été reçues. Si les données sont invalides, la couche MAC les ignorent et l'entité LLC n'est pas informée.
3. **MA-UNITDATA-STATUS.indication** permet à l'entité 802.15.4 de reporter le résultat de la dernière transmission de données. Ce rapport est automatiquement généré après un appel à **MA-UNITDATA.request**.

### 2.3.3 Structures de trames

802.15.4 propose 4 types de trames :

- données (**data**),
- acquittement (**ack**),
- balise (**beacon**),
- service (**MAC command**).

Ces quatre types de trames ont une structure commune : un champ contrôle de trame (*frame control*), un champ numéro de séquence (*sequence number*), un ou plusieurs champs propres au type de la trame puis une séquence de contrôle sur 16 bits (*Frame Check Sequence*).

#### 2.3.3.1 Le champ de contrôle de trame

Le champ de contrôle de trame a une longueur de 16 bits. Il indique :

- le type de trame,
- si la trame MAC est chiffrée ou non,
- si des données restent encore à envoyer au destinataire,
- si la trame envoyée nécessite un acquittement,
- si la trame est destinée au PAN (*intra-PAN*) ou à l'extérieur du PAN (*inter-PAN*),
- si les champs d'adressages source et/ou destination sont présents ou non,
- si les adresses passées sont courtes (16 bits) ou étendues (64 bits).

### 2.3.3.2 Le champ de numéro de séquence

Le champ de numéro de séquence a une longueur de 8 bits. Suivant le type de la trame, ce champ peut avoir des significations différentes. Dans le cas d'une trame balise, ce champ spécifie le numéro de séquence du *beacon* ( $BSN^\diamond$ ) ; ce numéro est choisi au hasard dans un premier temps par le coordinateur puis incrémenté à chaque balise. Dans le cas des trames de données, acquittement ou service, ce champ spécifie le numéro de séquence des données ( $DSN^\diamond$ ) ; ce numéro est propre à une communication d'une entité avec une autre. De même que pour le  $BSN$ , ce numéro est choisi au hasard puis incrémenté à chaque trame.

### 2.3.3.3 Le champ d'adressage

Le champ d'adressage est composé de 4 sous-champs :

- Identifiant de PAN destination : sur 16 bits. Si la valeur  $0xFFFF$  est passée (diffusion, *broadcast*), toutes les entités à l'écoute de ce canal considéreront le message,
- Adresse de destination : elle peut être sur 16 ou 64 bits, selon ce qui a été défini dans le champ de contrôle de trame. De même, si l'adresse courte  $0xFFFF$  est passée, toutes les entités à l'écoute de ce canal considéreront le message,
- Identifiant de PAN source,
- Adresse source.

Les identifiants de PAN (PAN-id, 16 bits) sont fixés au moment de la création du PAN par le coordinateur. En cas de conflit, c'est-à-dire si une entité du réseau détecte que deux coordinateurs utilisent le même PAN id, une procédure de résolution de conflit est déclenchée afin de changer l'identifiant. Enfin, notons que ces sous-champs peuvent être absents si l'absence d'adressage a été spécifiée dans le champ de contrôle de trame.

### 2.3.3.4 Le champ *payload*

Le champ *payload* (charge utile) contient les données utiles de la trame. Sa structure est différente selon le type de trame. Si le chiffrement a été précisé dans le champ de contrôle de trame, cette partie de la trame est chiffrée.

La trame balise est envoyée par un coordinateur d'étoile. Elle contient des informations sur la structure de la supertrame, sa durée effective, la durée de la période d'écoute obligatoire pour les entités synchronisées et le moment où commencera la suivante. Cette trame indique également si la demande d'association au PAN est possible ou non, si la réservation de créneaux temporels (GTS) est possible et leur nombre éventuel.

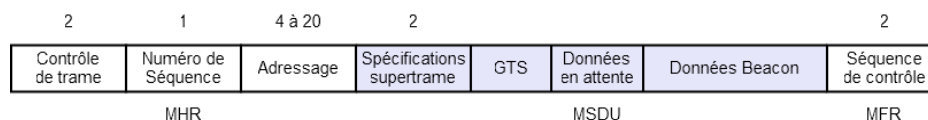


FIG. 2.12 – Structure de la trame balise

La trame de données contient les données passées par la couche supérieure, c'est-à-dire la couche réseau. La structure de sa charge utile (MSDU) interne dépend donc de cette couche.

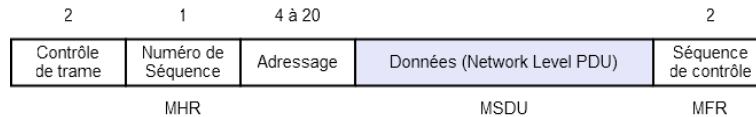


FIG. 2.13 – Structure de la trame de données

La trame d'acquittement est la seule qui ne contient pas de champ *payload*. L'identification de la trame qui est acquittée se fait grâce au numéro de séquence de la trame d'acquittement, qui est égal au DSN de la trame à acquitter.

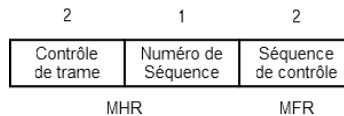


FIG. 2.14 – Structure de la trame d'acquittement

La trame de service permet de faire des requêtes auprès de la couche MAC des autres entités du réseau. Ces requêtes sont les suivantes :

- demande ou réponse d'association, notification de désassociation,
- demande de données,
- notification et réponse à un orphelin (perte du coordinateur),
- demande d'une balise,
- demande d'un GTS,
- notification de conflit de PAN ID.

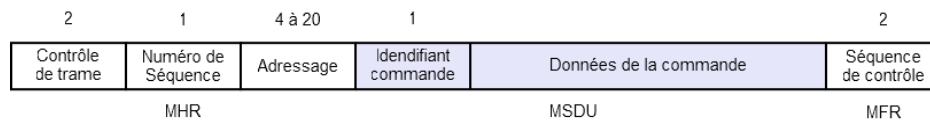


FIG. 2.15 – Structure de la trame de service

### 2.3.3.5 Le champ FCS

Le champ FCS (*Frame Check Sequence*) effectue la détection d'erreurs. Il est basé sur un CRC 16 bits ITU-T. Il prend en compte toute la trame de niveau 2, depuis le champ de contrôle de trame jusqu'à la charge utile (*payload*). Du fait de certaines trames (messages de gestion de niveau MAC) qui ne remontent pas au dessus de la sous-couche MAC, la vérification de l'intégrité des données doit être réalisée par la sous-couche MAC, alors que cette tâche est généralement laissée à la sous-couche de convergence située au dessus de la sous-couche MAC<sup>4</sup>.

<sup>4</sup>Cette particularité est très répandue dans les technologies de réseau qui nécessitent une gestion fine au niveau de la sous-couche MAC. Comme la méthode d'accès est basée sur un échange de *messages*, leur intégrité doit être impérativement vérifiée, ce qui ne peut être fait au niveau LLC...

## 2.4 La couche Réseau

Nous l'avions vu au début de ce chapitre, ZigBee recommande l'utilisation de la technologie proposée par le standard IEEE 802.15.4 pour les deux premières couches du modèle OSI (couche *physique* et couche *liaison*). Pour les couches supérieures, la ZigBee Alliance propose sa propre spécification [ZBAL 05], et notamment, sa propre couche *Réseau* (niveau 3 OSI). Bien que les travaux élaborés pendant la thèse portent essentiellement sur la méthode d'accès, et donc, sur la couche *liaison*, il est intéressant – et même nécessaire – de comprendre le fonctionnement de la couche 3 qui a un impact direct sur la façon dont la méthode d'accès va être sollicitée.

Évoquons tout d'abord les topologies possibles ayant une influence sur le travail de la couche 3.

### 2.4.1 Éléments de la topologie du réseau

ZigBee définit trois types d'éléments pour constituer un réseau.

1. Le Coordinateur ZigBee, ou *ZigBee Coordinator*, ZC :
  - il est unique pour tout le réseau ZigBee et il est à l'origine de la création du réseau,
  - il reprend les tâches du *PAN Coordinator* décrit dans le standard IEEE 802.15.4,
  - il agit comme simple routeur (ZR, voir point suivant) une fois le réseau créé.
2. Le Routeur ZigBee, ou *ZigBee Router*, ZR :
  - il doit d'abord s'associer avec le ZC ou un autre ZR,
  - il accepte que d'autres éléments du réseau s'associent à lui,
  - il reprend les tâches du coordinateur 802.15.4,
  - il relaie les messages selon un protocole de routage qui sera présenté plus bas,
  - il est optionnel.
3. Le nœud terminal, ou *ZigBee End Device*, ZED :
  - il doit d'abord s'associer avec le ZC ou un ZR,
  - il ne constitue qu'un élément final du réseau : il n'accepte ni association, ni participation au routage des messages.
  - il est lui aussi optionnel.

Ces trois types d'éléments sont très semblables à ce que propose le standard IEEE 802.15.4. Nous pouvons constater, comme nous l'évoquions en début de chapitre, que le réseau ZigBee peut avoir une portée plus étendue que le rayonnement d'un simple nœud 802.15.4 grâce au processus de routage qui permet le relais d'un message par une ou plusieurs entités jusqu'à la destination finale. Ce processus nécessite une connaissance du réseau qui est réalisée par un algorithme de découverte automatique que nous décrirons plus en détails par la suite.

#### 2.4.1.1 Topologie en arbre

Dans le cadre d'une topologie en arbre, le processus de création du réseau s'articule autour du ZC dont la mise en service marque le début de la phase de création. Par la suite, toutes les entités qui se trouvent à portée radio de ce nœud se rattachent à lui. Les entités qui ne sont pas à portée du ZC se rattachent à l'un des ZR, de proche en proche, jusqu'à la formation totale du réseau, comme l'illustre la figure 2.16. Sur l'exemple de la figure, le réseau se forme en trois temps. Cette formation hiérarchisée forme un arbre dont la racine est le ZC ; il convient de lier cette topologie à l'adressage et au routage adéquat, comme nous le verrons plus loin.



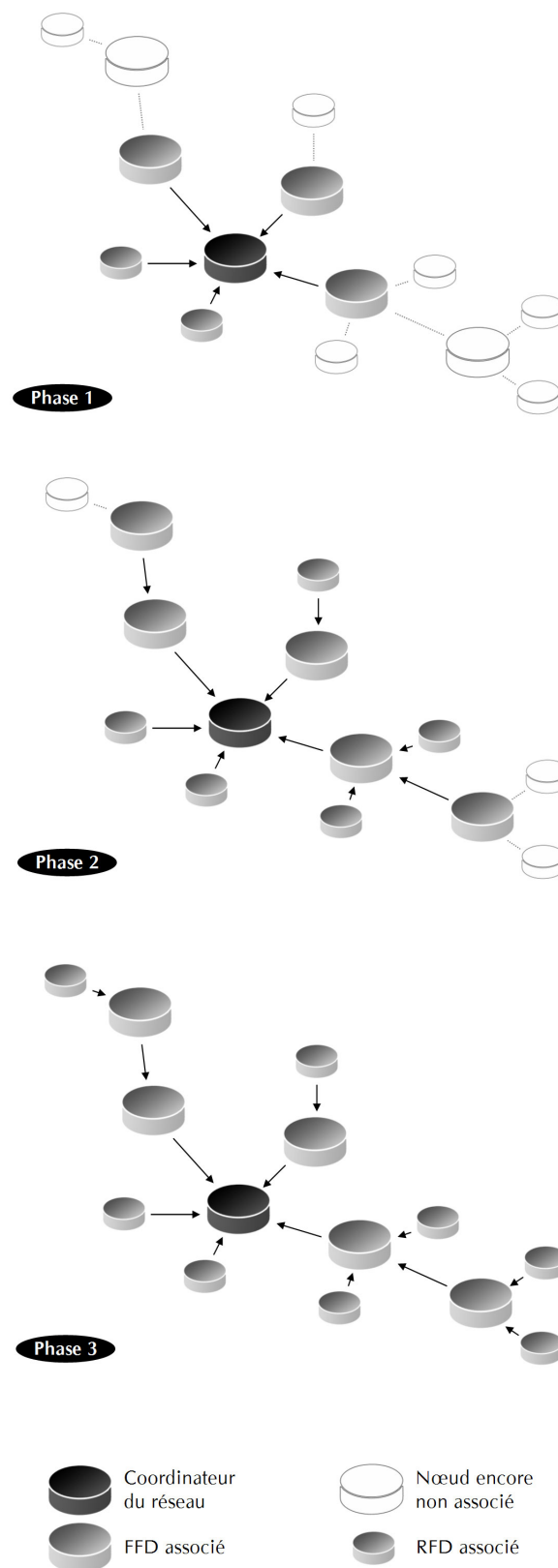


FIG. 2.16 – Exemple de création du réseau ZigBee en arbre

### 2.4.1.2 Topologie maillée

Avec la topologie maillée, ou *mesh*, tous les ZR à portée radio les uns des autres peuvent dialoguer entre eux, sans structure hiérarchique, comme l'illustre la figure 2.17. Un processus de routage doit être mis en place pour relayer les paquets de bout en bout du réseau ; nous verrons ce processus plus loin (cf. paragraphe 2.4.5).

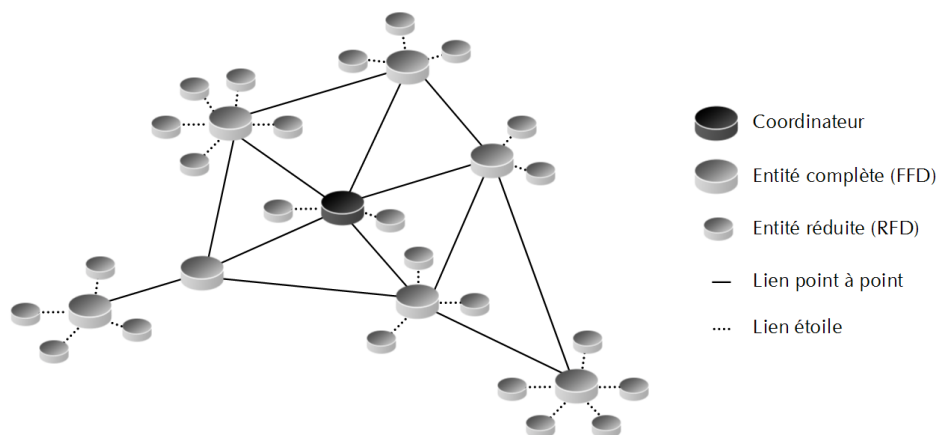


FIG. 2.17 – Exemple de topologie maillée

### 2.4.2 Architecture de la couche réseau

A l'instar de la couche liaison décrite plus haut par le standard IEEE 802.15.4, la couche réseau proposée par ZigBee se décompose en deux entités : une première entité (NLDE<sup>◇</sup>) traitant l'envoi, la réception des données et éventuellement la ré-émission de données reçues dans le cadre du routage, et une seconde entité (NLME<sup>◇</sup>) chargée de la gestion et de la maintenance du réseau (routage, adressage, etc.). L'entité NLDE communique avec la couche 2 via le point d'accès MCPS-SAP<sup>◇</sup> et met à disposition des couches supérieures son propre point d'accès NLDE-SAP<sup>◇</sup>. L'entité NLME communique avec la couche 2 via le point d'accès MLME-SAP<sup>◇</sup> et permet au ZDO<sup>◇</sup> de pouvoir communiquer avec elle via son point d'accès : NLME-SAP<sup>◇</sup>.

La couche réseau contient également une base d'informations appelée NIB<sup>◇</sup>. Elle contient les informations sur les nœuds voisins et la table de routage. La structure de la couche réseau proposée par ZigBee est représentée figure 2.18.

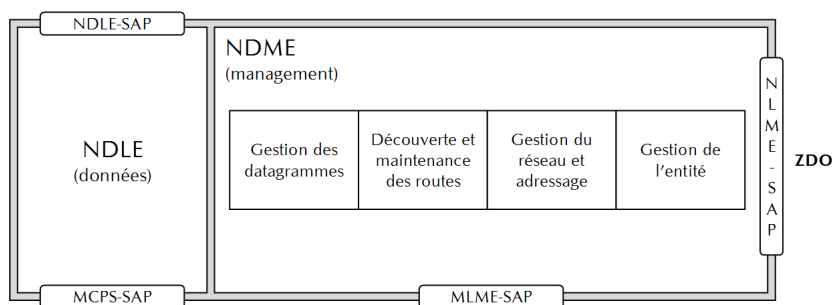


FIG. 2.18 – Structure de la couche réseau proposée par ZigBee

### 2.4.3 Services rendus

La couche Réseau de ZigBee est en charge des opérations suivantes :

- Création et traitement des PDU de niveau Réseau (NPDU<sup>◇</sup>),
- Participation au routage des paquets,
- Création (Coordinateur ZigBee) ou rattachement à un réseau existant,
- Adressage,
- Découverte du voisinage et stockage des informations liées aux nœuds voisins,
- Contrôle du récepteur, économie d'énergie (fonctions liées à la couche MAC).

### 2.4.4 Adressage

La spécification ZigBee autorise deux types d'adressage : un adressage libre laissé à la convenance de la couche supérieure (HHLE-based addressing<sup>◇</sup>) et un adressage de type hiérarchique en arbre (*Tree Structure*). Si l'adressage de type hiérarchique doit être utilisé, l'attribut `nwkUseTreeAddrAlloc` de la NIB devra être positionné à `TRUE`. Si, au contraire, c'est le mode d'adressage NHLE-based qui doit être utilisé, `nwkUseTreeAddrAlloc` devra être positionné à `FALSE`.

#### 2.4.4.1 Adressage NHLE-based

Si l'attribut `nwkUseTreeAddrAlloc` de la NIB est positionné à `FALSE`, une couche supérieure doit prendre en charge l'adressage et doit, pour cela, tenir compte de trois attributs de la NIB :

- `nwkNextAddress` indique la prochaine adresse qui sera attribuée,
- `nwkAvailableAddresses` est décrémenté à chaque nouvelle association,
- `nwkAddressIncrement` est la valeur de l'incrément qui doit modifier `nwkNextAddress` à chaque nouvelle association.

Si l'attribut `nwkAvailableAddresses` vaut 0, le ZR ne peut plus honorer de nouvelles demandes d'association et doit positionner son attribut `macAssociationPermit` à `FALSE`.

#### 2.4.4.2 Adressage en arbre

Si l'attribut `nwkUseTreeAddrAlloc` de la NIB est positionné à `TRUE`, les adresses sont distribuées suivant un algorithme hiérarchique ayant une structure arborescente. L'avantage de cette distribution est qu'elle est totalement décentralisée tout en garantissant l'unicité des adresses distribuées, car chaque routeur ZigBee dispose d'un certain nombre d'adresses qu'il peut distribuer aux nœuds qui s'associent à lui. Pour réaliser cette distribution d'adresses, chaque routeur doit connaître sa profondeur (notée *d*, pour *depth*, c'est-à-dire le nombre de sauts à effectuer pour atteindre le coordinateur), ainsi que trois autres paramètres :

- `nwkMaxDepth`, ou *Lm*, qui fixe la profondeur maximale du réseau ZigBee,
- `nwkMaxRouters`, ou *Rm*, qui fixe le nombre maximal de routeurs de profondeur *d* − 1 associés à un routeur de profondeur *d*,
- `nwkMaxChildren`, ou *Cm*, qui fixe le nombre maximal d'entités (ZR ou ZED) associées à un routeur.

La figure 2.19 illustre ces différents paramètres.

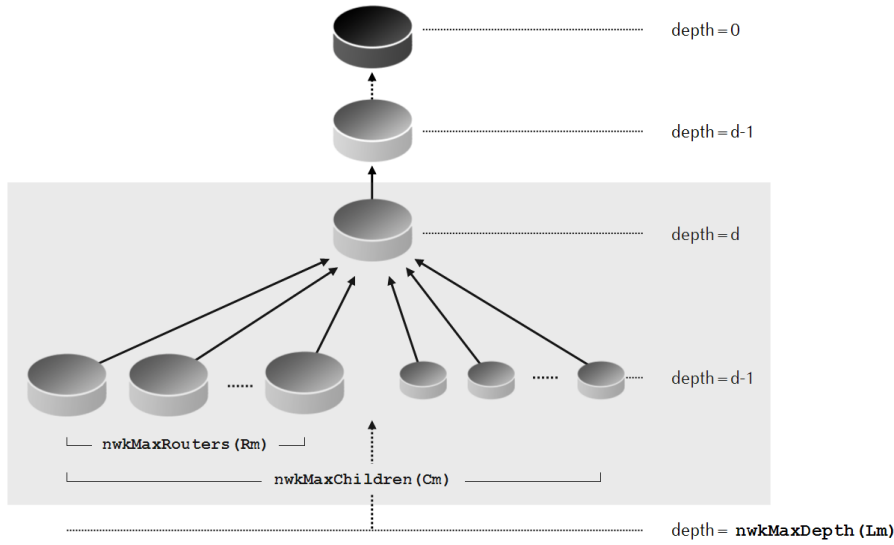


FIG. 2.19 – Illustration de la hiérarchie des adresses dans le cadre de l’adressage automatique en arbre

Connaissant ces quatre paramètres ( $d$ ,  $Lm$ ,  $Rm$  et  $Cm$ ), il est possible de calculer les différentes adresses des nœuds qui s’associent à un routeur. L’adresse  $A_n$  du  $n^{ime}$  nœud associé à un routeur de profondeur  $d + 1$  sera :

$$A_n = A_{parent} + Cskip(d) * Rm + n$$

avec

$$Cskip(d) = \begin{cases} 1 + Cm.(Lm - d - 1) & \text{si } Rm = 1 \\ \frac{1 + Cm - Rm - Cm.Rm^{Lm-d-1}}{1 - Rm} & \text{sinon} \end{cases}$$

Le principe de l’adressage en arbre est très intéressant car il permet d’une part de distribuer automatiquement des adresses de façon décentralisée et déterministe – il suffit de connaître les quatre paramètres sur lesquels se basent les calculs pour définir localement l’adressage cohérent sur tout le réseau – ce qui minimise les échanges de messages de gestion. Cette économie de messages échangés va dans le sens de l’économie d’énergie. D’autre part, et nous allons le voir plus bas, cet adressage permet un routage automatique en se basant sur les adresses ; là encore, nul besoin d’échange de messages pour déterminer quelle route emprunter ! En terme d’économie d’énergie, un réseau ZigBee a tout intérêt – si possible, par rapport à la géométrie des liens, les portées des nœuds et la présence d’obstacles – à utiliser ce type d’adressage car il va de pair avec une limitation des requêtes des demandes de route et une réduction de la mémoire utilisée pour le processus de routage.

#### 2.4.5 Principes de base du routage ZigBee

Du fait de la mémoire très limitée des nœuds, le routage ZigBee [MERC 04] [FRAN 06] suit le principe de base suivant :

- si la table de routage contient une entrée qui correspond au routage demandé, il faut router le paquet selon cette entrée,

- si elle ne contient aucune entrée et si la mémoire libre le permet, il faut lancer le processus de découverte de route,
- sinon, il faut router le paquet selon le routage hiérarchique en arbre (*Tree Routing*).

La structure de la table de routage d'un nœud ZigBee est représentée dans le tableau 2.3

Nom du champ	Taille	Description
Adresse de destination	2 octets	L'adresse courte du nœud de destination
Statut	3 bits	Le statut de la route
Adresse du prochain saut	2 octets	L'adresse courte de la passerelle

TAB. 2.3 – Structure de la table de routage d'un nœud ZigBee

#### 2.4.5.1 Algorithme de routage à la demande

L'algorithme de routage à la demande proposé par la ZigBee Alliance est proche d'AODV<sup>◇</sup> [PERK 97]. AODV est un protocole de routage purement « à la demande » c'est-à-dire que les nœuds qui participent au routage ne cherchent pas à maintenir des informations sur la topologie du réseau ou sur les routes.

Lorsqu'un nœud veut envoyer un message à un autre nœud et qu'il ne possède pas l'entrée correspondante dans sa table de routage, il lance le processus de recherche de chemin (*Path Discovery*) en diffusant une requête de recherche de route (**route\_request**). Cette requête est relayée par les routeurs voisins, et ainsi de suite, par inondation sur tout le réseau. Lorsque le nœud de destination reçoit la requête, il répond en envoyant un message de réponse (**route\_reply**) mais à la différence du **route\_request** qui avait été envoyé par diffusion, la réponse est envoyée seulement au dernier routeur ayant relayé la diffusion et ainsi de suite, jusqu'au nœud qui avait initialement demandé la route. Pour retrouver le chemin inverse, chaque routeur rediffusant un message **route\_request** doit mémoriser par quel voisin ce message lui est parvenu.

#### 2.4.5.2 Algorithme de routage hiérarchique : Tree Routing

Le cas particulier de la topologie en arbre permet de simplifier grandement le routage puisque, dans le cadre de cette topologie hiérarchique, l'adressage est fonction de la topologie. Il suffit donc d'observer l'adresse du nœud de destination pour déterminer à quel voisin envoyer le paquet.

L'adresse du routeur voisin est déterminée ainsi : soit  $A$  l'adresse du ZR, et  $D$  l'adresse de destination,

- si l'inégalité  $A < D < A + Cskip(d - 1)$  est fausse, alors la destination est située « plus haut », c'est-à-dire vers la racine de l'arbre, donc vers le coordinateur ZigBee ; il faut donc router le paquet en le transmettant à son nœud « père ».
- sinon, la destination est située « plus bas », c'est-à-dire dans sa propre « progéniture ». Deux cas peuvent alors se présenter :
  - si le ZR n'a qu'un seul « fils », il route le paquet en l'envoyant à ce dernier,
  - si le ZR a plusieurs « fils », il route le paquet en l'envoyant au « fils » d'adresse

$$A + 1 + \frac{D - (A + 1)}{Cskip(d)} \cdot Cskip(d)$$

Le problème principal de cet algorithme de routage est que, de part sa structure arborescente, il ne permet pas la redondance des liens comme dans un réseau *mesh*. Si un lien vient à tomber, c'est toute une partie du réseau qui se retrouve isolée. Conceptuellement, sa fiabilité paraît plus faible que celle d'un réseau *mesh*. De plus, il n'est pas toujours réalisable facilement, par exemple dans le cas d'un réseau assez mobile.

#### 2.4.6 Ordonnancement des beacons dans un réseau avec arborescence

Dans le cadre d'une topologie multi-sauts, il est nécessaire d'ordonner l'émission des trames beacons afin d'éviter que celles-ci n'entrent en collision avec d'autres trames beacons<sup>5</sup>. Dans sa spécification, ZigBee propose un mécanisme d'évitement de collisions de beacon en répartissant les portions actives des supertrames dans le temps pour des ZR se trouvant dans le même voisinage.

Pour ce faire, les paramètres *BO* et *SO* doivent être différents et, si possible, *BO* doit être le plus grand possible<sup>6</sup> devant *SO*. Le temps est alors découpé en portions de durée  $\frac{2^{BO}}{2^{SO}}$  et chaque ZR du voisinage doit placer la portion active de sa supertrame (qui commence par un beacon) dans l'une de ces portions. Le délai qui affecte l'émission des beacons est indiqué dans le beacon émis. La connaissance de cette information permet d'une part de déterminer le moment où sa propre entité « père » reçoit le beacon de son entité supérieure (*le beacon de son propre « grand-père »*, en quelque sorte!), mais aussi le moment où chaque nœud du voisinage reçoit le *beacon* de son parent. En ayant connaissance de toutes ces informations et aussi de la valeur de *SO*, chaque nœud du réseau peut éviter de provoquer des collisions de beacons, mais aussi plus généralement des « collisions de portions actives de supertrames ».

Notons cependant qu'un nœud doit nécessairement être actif dans la portion active de son « père » (écoute des beacons, transfert de données indirect, etc.).

#### 2.4.7 Structure du paquet de niveau réseau

Nous concluons cette partie sur le niveau *réseau* de ZigBee par la structure du paquet de niveau 3, comme l'illustre la figure 2.20. Sur cette figure, on peut également voir comment le paquet de niveau 3 est encapsulé dans la trame DATA de niveau 2 (802.15.4).

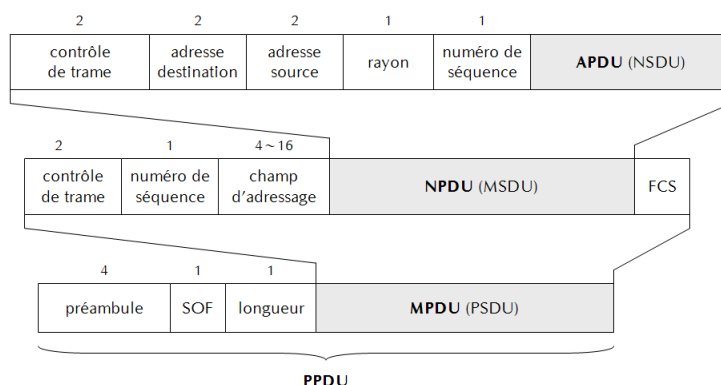


FIG. 2.20 – Structure du paquet de niveau *réseau* et encapsulation dans une trame de données 802.15.4

Comme la trame 802.15.4, le paquet ZigBee possède un champ de contrôle et un champ de numéro de séquence propre, ainsi qu'un champ d'adressage (source, destination) pour l'adressage de bout-en-bout

<sup>5</sup>Cet ordonnancement est nécessaire dans le cas d'une topologie multi-sauts avec arborescence mais pas dans le cas d'une topologie multi-sauts maillée, puisque dans ce cas précis, l'émission régulière de beacons n'est pas autorisée.

<sup>6</sup>Le plus grand possible, autant que l'application le permet...

du réseau. Notons également la présence du champ *rayon* qui est modifié à chaque passage dans un routeur (similaire au *Time-To-Live* d'IP).

### 3 Identification des failles dans la méthode d'accès proposée par le standard

Après avoir présenté les trois premiers niveaux (*physique, liaison et réseau*) de 802.15.4/ZigBee, nous détaillons à présent les différentes failles que nous avons identifiées dans l'ensemble de ces protocoles, au regard de notre objectif principal de *transport d'informations à fortes contraintes temporelles*.

#### 3.1 La gestion des slots garantis ne présente pas une garantie absolue

Comme nous l'avons évoqué précédemment, IEEE 802.15.4 propose un mécanisme de réservation du médium qui permet à quelques nœuds privilégiés, dans une phase particulière du protocole, d'être épargnés par le phénomène des collisions (GTS). Cependant, ce mécanisme n'est pas parfait car l'obtention d'une réservation du médium est conditionnée par deux points :

1. Le réseau ne doit pas être saturé ; effectivement, sa capacité n'est pas infinie. Mais la norme ne fournit aucune possibilité au coordinateur d'une étoile de « réserver a priori » une part de la bande passante<sup>7</sup> à certains nœuds que l'on pourrait qualifier de « critiques ». Les premiers demandeurs sont les premiers servis, ce qui est une politique d'ordonnancement de type *Best-Effort*, incompatible avec l'idée de Qualité de Service [BOUY 97].
2. L'appel de la primitive de demande de réservation de bande passante (**GTS.request**) génère une trame qui est envoyée au coordinateur en utilisant le protocole CSMA/CA. Comme nous l'avons vu dans le premier chapitre dans le cadre de 802.11PCF, ce protocole fonctionne en *Best-Effort* et il ne permet donc pas d'avoir de certitude sur le plan temporel, d'une part à cause du délai sur lequel repose son algorithme, mais aussi d'autre part à cause des collisions, toujours probables puisque l'algorithme n'est pas déterministe. Comme l'illustre la figure 2.21, les **GTS.request**, comme le reste du trafic CSMA/CA, peuvent être reportés sans garantie d'accès au médium.

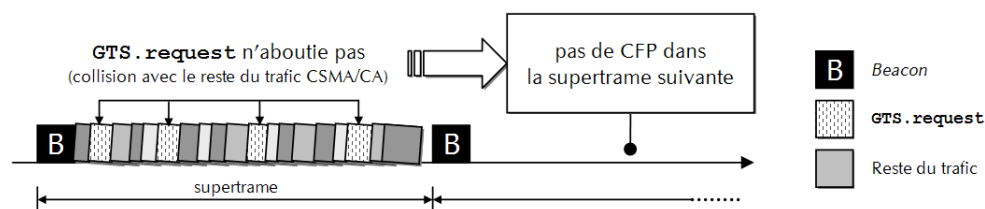


FIG. 2.21 – Les messages **GTS.request** peuvent entrer en collision avec le reste du trafic CSMA/CA

Par extension, le mécanisme d'association au réseau, qui utilise la primitive **association.request**, nécessite l'envoi de messages soumis aux aléas inhérents à cet algorithme ; là encore, aucune certitude n'est possible quant au temps nécessaire pour s'associer au réseau, réserver de la bande passante, et transmettre ses données.

<sup>7</sup>Nous appelons ici *bande passante* le débit offert par la couche 2.

### 3.2 Autres problèmes identifiés

Dans le cadre d'une utilisation de 802.15.4 dans un réseau de capteurs très contraints temporellement, il serait intéressant de pouvoir garantir un débit et un délai de transmission pour certains nœuds critiques pour l'application communicante. De plus, le réseau peut être amené à prendre en charge des trafics très différents : forts débits sporadiques, faibles débits très contraints temporellement, très faibles débits réguliers issus de nœuds contraints énergétiquement.

Ainsi, d'autres faiblesses liées à l'utilisation des GTS apparaissent :

- Un nœud, s'il arrive à s'associer au réseau et à obtenir la bande passante qu'il désire dans le temps imparti, ne peut cependant pas conserver son acquisition tant qu'il le souhaite. En effet, dans l'état actuel de la norme, une suite de GTS est attribuée pour un nombre fixé de supertrames (dépendant de *BO*), et ne peut pas être maintenue sans reprendre entièrement le processus de requête (nouvelle génération d'une *GTS.request* en *Best-Effort*, etc.). Il n'est pas possible d'assurer la continuité du service garanti car le message *GTS.request* ne peut être envoyé dans l'un des derniers GTS attribué. La figure 2.22 illustre ce défaut où, après avoir bénéficié d'une série de GTS, un nœud ne peut pas accéder au médium trop sollicité et perd son accès déterministe.

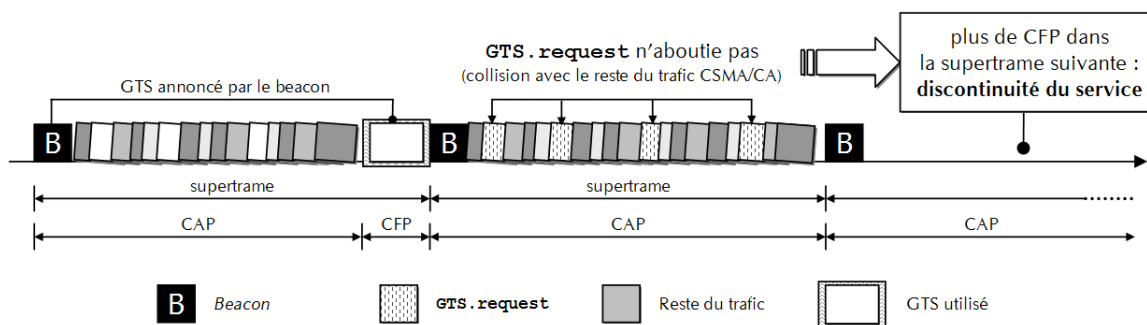


FIG. 2.22 – Le service garanti (GTS) ne peut être maintenu de manière déterministe

- Si il est possible d'affecter plusieurs GTS à un même nœud dans la même supertrame (donc d'augmenter le débit dédié à ce nœud), il n'est en revanche pas prévu de pouvoir affecter un GTS toutes les deux supertrames, voire encore moins fréquemment (diminuer le débit réservé mais conserver un accès déterministe et régulier au médium). Les GTS reviennent périodiquement avec la même fréquence : celle de la supertrame. Pour faire changer cette fréquence, il convient de changer les paramètres *BO* et *SO*, donc faire varier l'horloge de tous les transferts de données de l'étoile. Il n'est pas possible, par exemple, de prévoir un GTS toutes les deux supertrames ; la bande passante est alors perdue, comme l'illustre la figure 2.23. Autrement dit, dans l'état actuel de la norme, il n'est pas prévu, dans le mode sans contention, de pouvoir faire cohabiter des trafics très différents les uns des autres.



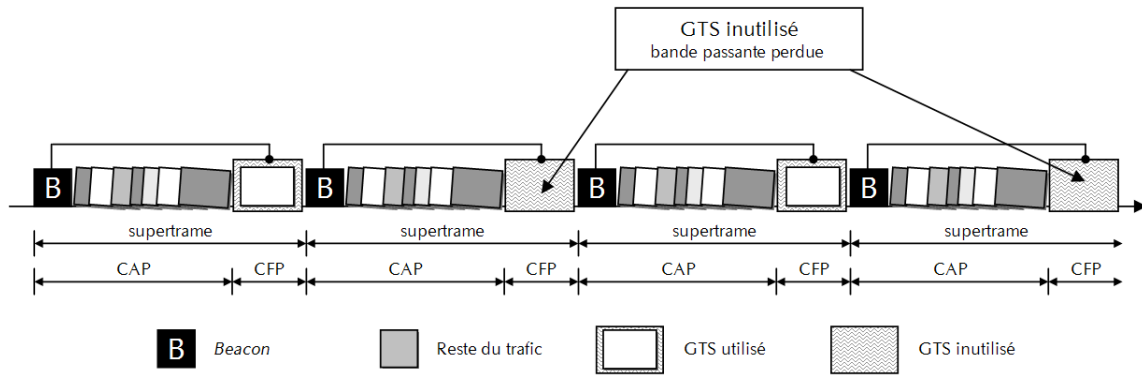
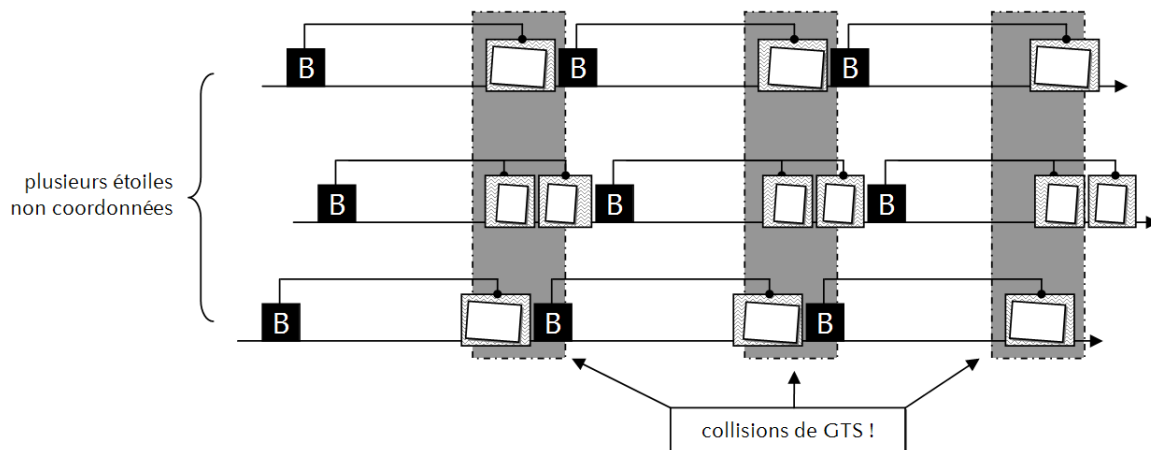


FIG. 2.23 – Un GTS inutilisé constitue une perte de bande passante non réutilisable

- Si plusieurs étoiles cohabitent dans la même zone de portée radio et sur le même canal, il y a risque de collisions pour des trafics utilisant des GTS car la norme ne prévoit pas de mécanisme de communication entre deux coordinateurs d'étoile (cf. figure 2.17 page 72, liens pleins) qui décident, chacun de leur côté, d'attribuer des GTS à leurs nœuds. Dans ce cas, on peut parler de *collisions de GTS*, comme l'illustre la figure 2.24.


 FIG. 2.24 – Plusieurs coordinateurs attribuant des GTS peuvent provoquer des *collisions de GTS*

### 3.3 Conclusion : les améliorations proposées

A la lumière de ce qui vient d'être évoqué, le mécanisme de réservation du médium pourrait être amélioré pour permettre le transport de flux à très fortes contraintes temporelles. Les principes de base peuvent être conservés mais deux points doivent être optimisés :

- La distribution et la répartition des GTS doivent gagner en souplesse, le mécanisme doit pouvoir être plus dynamique pour permettre au réseau de prendre en charge des flux de physionomies différentes,
- Les coordinateurs ou les ZR qui cohabitent dans la même zone de portée radio et sur le même canal radio doivent pouvoir s'échanger des informations sur la répartition des GTS afin de limiter les *collisions de GTS dans la CFP*. Le mécanisme d'ordonnancement des beacons prévu par ZigBee (cf. section 2.4.6 de ce chapitre, page 76) constitue un point de départ intéressant dans la mesure où les collisions de beacons sont rendues plus rares mais, en revanche, il ne règle pas le problème des *collisions de GTS*.

Ainsi, nous proposons de renforcer ce mécanisme par la mise en œuvre d'une méthode d'accès au médium totalement déterministe, comme cela peut être requis dans le cadre d'un réseau de capteurs et, plus largement, dans les réseaux industriels sans fil. La gestion des accès déterministes doit également être très souple, afin de rendre possible la cohabitation de flux à physionomies radicalement différentes.

Dans le chapitre suivant, nous présentons les principes et les améliorations que nous proposons dans ce travail, ainsi qu'une application typique pour illustrer les besoins.

# Bibliographie du chapitre 2

- [BOUY 97] E. BOUYER et E. HORLAIT – *Bandwidth Management and Reservation over Shared Media – Segundo Seminário Franco-Brasileiro em Sistemas Informáticos Distribuídos (SFBSID'97)*, Fortaleza, Brazil (1997)
  - [CALL 02] E. CALLAWAY, P. GORDAY, L. HESTER, J.A. GUTIERREZ, M. NAEVE, B. HEILE et V. BAHL – *Home Networking with IEEE 802.15.4 : a developing standard for low-rate wireless personal area networks* – IEEE Communications Magazine vol. 40, no. 8, pp 70-77 (Août 2002)
  - [FRAN 06] J. FRANCOMME, G. MERCIER et T. VAL – *A simple method for guaranteed deadline of periodic messages in 802.15.4 cluster cells for automation control applications – IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'2006)*, Prague. Czech Republic (Septembre 2006)
  - [FRE1 04] FREESCALE SEMICONDUCTOR – *ZigBee / IEEE 802.15.4 FREESCALE solution* – Technology General Information (2004)
  - [FRE2 04] FREESCALE SEMICONDUCTOR – *ZigBee Reference Design (ZRD01) documentation & applications notes* (2004)
  - [FRE3 04] FREESCALE SEMICONDUCTOR – *MC13192 Reference Manual* (2004)
  - [FRE4 04] FREESCALE SEMICONDUCTOR – *802.15.4 MAC/PHY Software User's Guide* (2004)
  - [GUIL 05] C. GUILLEMINOT, L. ANDRIEUX et J.J. MERCIER – *A contribution to a virtual prototyping for a spread spectrum telecommunication system using VHDL-AMS – BMAS'05*, San Jose, Californie, USA (Septembre 2005)
  - [IEE1 99] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.11 IEEE Standard for Information technology, Specific Requirements – Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications* – IEEE standard 802-11 (1999)
  - [IEE2 03] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.15.4 IEEE Standard for Information technology, Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs) – IEEE Std 802.15.4-2003* (2003)
  - [KHOU 05] T. KHOUTAIF et F. PEYRARD – *Performances evaluation of the asynchronous Bluetooth links in a real time environment – IEEE Personal Wireless Communications (PWC'2005)*, Colmar, France (Août 2005)
  - [LLI1 06] J.F. LLIBRE, P. PINEL et E. CAMPO – *Dimensionnement d'un générateur photo voltaïque pour un système communicant autonome – XII<sup>ème</sup> Colloque National de la Recherche dans les IUT*, Brest, France (2006)
  - [MAXS] Le site web de MAXSTREAM – <http://www.maxstream.net>
  - [MERC 04] G. MERCIER – *Un réseau sans fil faible consommation : IEEE 802.15.4 (alias ZigBee)* (2004)
  - [PERK 97] C. PERKINS et E. ROYER – *Ad-hoc on-demand distance vector routing – MILCOM'97* (Novembre 1997)
  - [VDB1 05] A. VAN DEN BOSSCHE, T. VAL et E. CAMPO – *Métrie pour l'analyse comparative des performances temporelles des liens Bluetooth – IEEE Sciences of Electronic, Technologies of Information and Telecommunication (SETIT'2005)*, Sousse, Tunisie (Mars 2005)
-

- [ZBAL 05] ZIGBEE ALLIANCE – *ZigBee Specification* – ZigBee Document 053474r06, version 1.0 (2005)
- [ZBAL] Le site web de la ZIGBEE ALLIANCE – *www.zigbee.org*
- [ZHEN 04] J. ZHENG et M. J. LEE – *Will IEEE 802.15.4 make ubiquitous networking a reality ? : A discussion on a potential low power, low bit rate standard* – IEEE Communications Magazine, vol. 27, no. 6, pp. 23-29 (2004)

## Chapitre 3

# Vers un WPAN à Qualité de Service pour des transports d'informations à fortes contraintes temporelles

*Après avoir détaillé les normes IEEE 802.15.4/ZigBee et identifié certaines failles sur la méthode d'accès au médium, ce chapitre expose notre contribution au transport d'informations à fortes contraintes temporelles sur un réseau personnel sans fil (WPAN). Dans un premier temps, et afin de situer le travail dans son contexte, nous présenterons une application typique de robotique communicante sur laquelle a été déterminé le cahier des charges, même si la proposition peut s'appliquer à tout type de réseau sans fil personnel. Dans un second temps, nous présenterons le protocole qui a été proposé pour répondre aux exigences de QoS et d'économie d'énergie.*

---

<b>1</b>	<b>Une application typique de robotique mobile communicante sans fil . . .</b>	<b>85</b>
1.1	Problématique générale : pourquoi introduire des communications sans fil pour des robots? . . . . .	85
1.2	Les attentes vis-à-vis d'un système de communication adapté . . . . .	86
1.2.1	Un système proposant des garanties sur le plan temporel . . . . .	86
1.2.2	Retransmissions vs. redondance et codes correcteurs d'erreurs . . . .	86
1.2.3	Taille du réseau . . . . .	87
1.2.4	Topologie du réseau . . . . .	87
1.2.4.1	Communications locales au robot . . . . .	87
1.2.4.2	Communications entre robots . . . . .	87
1.2.5	Débit nécessaire . . . . .	88
1.2.6	Conclusion . . . . .	89
<b>2</b>	<b>La proposition protocolaire . . . . .</b>	<b>90</b>
2.1	Nouvelles fonctionnalités proposées . . . . .	90
2.2	Problématique de la cohabitation de plusieurs étoiles sur un médium commun	91
2.3	Étude du protocole mis en œuvre . . . . .	92
2.3.1	Éléments du réseau, liens et portées . . . . .	92
2.3.2	Organisation temporelle de l'accès au médium . . . . .	93
2.3.3	Mécanisme de demande de réservation du médium . . . . .	94
2.3.4	Allocations au préalable : arrivées déterministes dans le réseau . . .	94
2.3.5	Gestion des acquittements . . . . .	95
2.3.6	Gestion et diffusion des informations de réservation . . . . .	96
2.3.7	Un exemple de demande de réservation du médium . . . . .	96
2.3.8	Une politique d'accès déterministe par défaut . . . . .	99

2.3.9	Optimisation de l'accès au médium : attribution de GTS simultanés	100
2.3.9.1	Le principe . . . . .	100
2.3.9.2	Nécessité d'une négociation protocolaire . . . . .	100
2.3.9.3	Application à notre topologie . . . . .	102
2.4	Implémentation du protocole : intégration dans une pile IEEE 802.15.4 . . . .	104
<b>3</b>	<b>Conclusion</b> . . . . .	<b>105</b>

---

# 1 Une application typique de robotique mobile communicante sans fil

Dans cette partie, nous présentons une application typique de robotique mobile dont les besoins en terme de communication nous permettent de définir un cahier des charges pour notre réseau.

## 1.1 Problématique générale : pourquoi introduire des communications sans fil pour des robots ?

Dans une application de robotique communicante, les robots communiquent, soit avec une ou plusieurs entités de commande, soit avec d'autres robots. On parlera dans ce dernier cas de *robots coopérants*. Notre application prend place dans ce domaine de la robotique mobile.

Notre application comprend deux objectifs :

1. d'une part, la suppression des nombreux câblages internes aux robots mobiles autonomes, ceci dans le but de faciliter leur fabrication et leur maintenance, de les alléger, et de pouvoir rapidement les réparer ou les modifier sans câblage complexe (capteurs interchangeables, temporaires),
2. d'autre part, la communication des robots entre eux au travers d'un même canal de communication.

Ce dernier point sur l'unicité du canal de communication est fondamental. En effet, d'un point de vue protocolaire, si le système de communication utilise plusieurs canaux et un seul récepteur par robot (un seul canal accessible à un instant  $t$ ), il devient très complexe de suivre simultanément sans perte de temps toutes les communications de type *point/multipoints*<sup>1</sup>. De plus, si le système peut fonctionner sur un canal unique, rien n'empêche ensuite de développer un applicatif utilisant plusieurs canaux.

D'un point de vue applicatif, l'unicité du canal de communication est très intéressante dans le cadre de robots coopérants car les informations issues des capteurs internes à un robot peuvent être récoltées à tout moment par tous les autres robots, y compris de manière passive si les capteurs du robot n°1 peuvent émettre jusqu'au robot n°2. Si l'on considère simplement deux robots coopérants qui évoluent dans un même milieu, chaque robot pourra alors profiter des capteurs de son associé. La faculté ainsi développée qui consiste à « voir dans les yeux de l'autre » est intéressante car les robots reçoivent alors plus d'informations et peuvent accéder à une meilleure connaissance du milieu dans lequel ils évoluent. Si l'un des robots subit une panne liée à un environnement hostile (chaleur excessive, chute dans une cavité, etc.) et s'il est équipé d'un capteur capable de détecter un tel événement, les autres robots de la flotte sont immédiatement prévenus. Ils peuvent ainsi prendre connaissance du danger et réagir en conséquence par anticipation avant même que leurs propres capteurs ne détectent le problème. La fonctionnalité de « voir dans les yeux de l'autre » se justifie d'autant plus que les robots mobiles sont par définition des entités embarquées, donc contraintes énergétiquement. La multiplicité des voies d'acquisition de données constitue un moyen intéressant d'économiser de l'énergie au niveau applicatif puisqu'une information acquise par un capteur d'un autre robot de la flotte coopérante n'est plus à acquérir.

De plus, de nombreux roboticiens sont freinés par la lourde et inévitable complexité inhérente à la pause d'un nouveau capteur sur un robot, surtout si ce robot est un prototype en constante évolution. Le développement d'un capteur totalement autonome (sans fil, y compris en alimentation [LLI2 06]), de petite taille, avec une très grande facilité de fixation (type aimant sur un carter) donnerait plus de flexibilité à la construction de robots multi-sensoriels.

---

<sup>1</sup>C'est à dire des communications dont l'émetteur envoie des messages à destination de plusieurs points comme dans le cadre d'une diffusion générale (*broadcast*) ou d'une diffusion par classes (*multicast*).

## 1.2 Les attentes vis-à-vis d'un système de communication adapté

### 1.2.1 Un système proposant des garanties sur le plan temporel

Notre application nécessite un support de communication très fiable ; les ordres transmis, c'est-à-dire les données acheminées par le réseau, doivent bien sûr arriver sans erreur à leur destinataire, mais aussi *à temps*. Le système de communication doit donc intégrer des propriétés de Qualité de Service (QoS<sup>◇</sup>), pour pouvoir proposer des garanties sur le transport de l'information, notamment sur le délai d'acheminement et sur les pertes des paquets comme le proposent certaines technologies de *bus de terrain* et/ou Réseaux Locaux Industriels [THOM 05]. Cependant, nous l'avons vu dans le premier chapitre de cette thèse, de nombreuses technologies WLAN<sup>◇</sup>/WPAN<sup>◇</sup> proposent des mécanismes de QoS, mais ceux-ci sont généralement incomplets quand ils sont disponibles matériellement [VAL 05].

D'un autre côté, il faut garder à l'esprit que toute technologie de transmission utilisant les ondes hertziennes sera soumise à de nombreuses perturbations et à un affaiblissement significatif du signal avec la distance ou les obstacles. La mobilité des robots entraîne également des problématiques car le réseau est mobile (problématiques de niveau physique : effet *doppler*, adaptation du récepteur, etc. mais aussi problématiques de niveau réseau : routage, etc.). Toutes ces contraintes imposées par la nature et par l'application limiteront fortement et dans tous les cas la portée et les débits d'une transmission réalisée sur le médium immatériel.

### 1.2.2 Retransmissions vs. redondance et codes correcteurs d'erreurs

Proposer des garanties temporelles sur un réseau sans fil, qui de plus est basé sur un médium partagé, n'est pas chose facile. Les perturbations ponctuelles qui affecteront la transmission pourront occasionner des erreurs, qui, si elles sont détectées, donneront lieu à des retransmissions. La ré-émission des données est nécessaire pour assurer l'intégrité de la transmission.

Cependant, les données ré-émises :

- diminuent le débit efficace de la transmission (même message envoyé à plusieurs reprises),
- augmentent le délai de transmission, ce qui impacte la latence globale du transport de l'information,
- occasionnent des pertes d'énergie puisque les données sont retransmises une nouvelle fois.

De plus, un système de fiabilisation par retransmission nécessite une gestion de messages *d'acquiescement* qui contribuent à la surcharge du canal et à l'augmentation des délais pour la communication globale : pour certaines applications de réseaux de capteurs, la durée de validité d'une information est si courte que l'obsolescence du message est certaine dès la première retransmission. De part la nature des données véhiculées (données issues de capteurs/actionneurs qui imposent de fortes contraintes temporelles) et le caractère embarqué (qui impose de fortes contraintes énergétiques) le système de communication adapté à une application de robotique mobile devra donc impérativement proposer des mécanismes de gestion du médium qui limitent le phénomène des retransmissions.

Fort heureusement, les mécanismes de retransmissions ne sont pas les seules solutions pour fiabiliser une transmission de données. Les *codes correcteurs d'erreurs* constituent également un moyen pour fiabiliser un transport d'informations, mais cette fois-ci sans augmenter exagérément la latence de transmission, si ce n'est par le volume des données à transmettre plus important<sup>2</sup>. Les techniques de corrections par codes correcteurs sont nombreuses (codage canal, FEC, codes de *Hamming*, codes convolutionnels, *turbocodes*, *Reed Salomon*, etc.). De plus, les avancées technologiques dans le domaine de la micro-électronique permettent d'intégrer les algorithmes de détection/correction d'erreurs directement dans le silicium, intégration synonyme de performances sur les plans temporel et énergétique. Cependant, les codes correcteurs présentent un inconvénient : ils sont consommateurs de bande passante

---

<sup>2</sup>Avec un système de fiabilisation des données basé sur les retransmissions, le risque est d'avoir une gigue de transmission importante. Avec un système basé sur des codes correcteurs, toutes les transmissions seront légèrement plus longues (plus de données à transmettre : temps de sérialisation et temps de traitement) mais la gigue sera théoriquement plus faible.



mais, a priori ce dernier point n'est pas gênant dans notre application où le volume des informations échangées reste très faible (comme dans tout réseau industriel de capteurs).

D'un point de vue applicatif, il est également possible de doubler les envois de messages (*suréchantillonnage*) pour palier aux éventuelles pertes dues à un taux d'erreur bit trop important.

### 1.2.3 Taille du réseau

L'application proposée requiert un réseau relativement « petit » et de dimension « finie » : d'une part, le nombre d'entités communicantes est faible – on pourra considérer qu'il n'y aura pas plus de 20 nœuds connectés et la phase de *passage à l'échelle* tiendra compte de cette hypothèse – et d'autre part, toutes les entités sont connues à l'avance.

### 1.2.4 Topologie du réseau

Comme nous l'avons évoqué plus haut, l'objectif du réseau de communication dans cette application est double : d'une part, de permettre la communication entre capteurs, actionneurs et unité de contrôle/commande d'un même robot et d'autre part, de permettre à une flotte de robots coopérants de communiquer entre eux, voire avec l'extérieur (une unité de télécommande, de gestion/administration ou bien encore une passerelle vers un autre réseau comme Ethernet ou même Internet).

#### 1.2.4.1 Communications locales au robot

Concernant le premier objectif – les communications locales à un robot – c'est naturellement une topologie en étoile qui est proposée, étoile qui a pour centre l'unité de contrôle/commande du robot et autant de branches que le robot compte de capteurs et d'actionneurs. La topologie en étoile convient parfaitement à notre application car avec cette topologie, tous les messages passent par le point central, ce qui correspond bien au cheminement naturel des données dans le cadre de notre application : on voit mal en effet, dans notre application, comment un capteur pourrait prendre l'initiative d'envoyer un ordre à un actionneur sans passer par le centre de contrôle/commande. Même dans un cas d'extrême urgence où une information captée nécessite une intervention immédiate (freinage soudain, changement de direction pour éviter un obstacle imprévu, etc.), le capteur doit nécessairement informer le centre de contrôle car celui-ci est le seul à connaître tous les éléments nécessaires pour décider de l'action à mener (vitesse de déplacement, présence ou non d'obstacles latéraux ou arrières, etc.). En revanche, l'exemple montre qu'il est absolument nécessaire de pouvoir compter sur un système de communication très fiable, tant sur l'intégrité des données que sur leur temps d'acheminement par le réseau, notamment pour des messages qui pourraient être considérés comme critiques pour le robot. L'exemple montre aussi que tous les messages multiplexés ne sont pas égaux sur le plan de la priorité : le système doit donc être en mesure de pouvoir plusieurs classes pour pouvoir rendre certains messages plus prioritaires.

Le centre de contrôle/commande jouera le rôle de coordinateur d'étoile ( $\text{FFD}^\diamond$ ,  $\text{ZR}^\diamond$ ), les capteurs/actionneurs seront des nœuds terminaux ( $\text{RFD}^\diamond$ ,  $\text{ZED}^\diamond$ ). La figure 3.1 illustre la topologie du réseau proposé pour la communication interne au robot ou charriot.

#### 1.2.4.2 Communications entre robots

Concernant le second objectif – les communications entre robots – c'est une topologie en point à point qui est proposée. Les robots entrent directement en communication les uns avec les autres, selon les besoins de coopération. Chaque centre de contrôle/commande peut établir un lien point à point avec un autre centre de contrôle/commande et router une information issue de l'un de ses capteurs vers

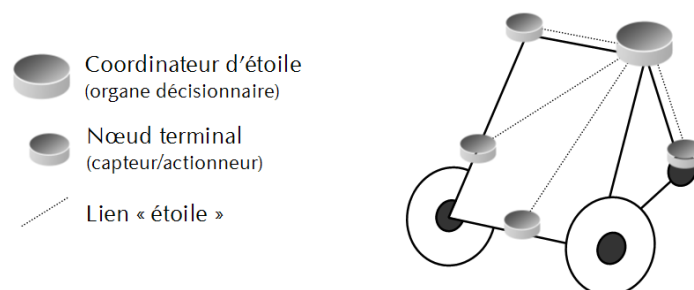


FIG. 3.1 – Topologie proposée pour la communication interne au robot

les autres robots ou interpréter un message issu de l'un de ses collaborateurs. Éventuellement, si deux robots désireux de communiquer ensemble ne sont pas à portée radio l'un de l'autre, un dispositif de routage assurera la connectivité.

Ajoutons que dans le cadre de notre application, il est également possible de rencontrer un autre type d'entité : un coordinateur dépourvu de nœud terminal, soit parce que ses nœuds sont tous inactifs pour une durée prolongée, soit parce qu'il n'est lié à aucun capteur ou actionneur. Cette entité peut par exemple être une entité fixe comme une balise de localisation, une passerelle vers un autre réseau (Internet...).

La figure 3.2 illustre la topologie du réseau proposé pour la communication entre les robots de la flotte coopérante.

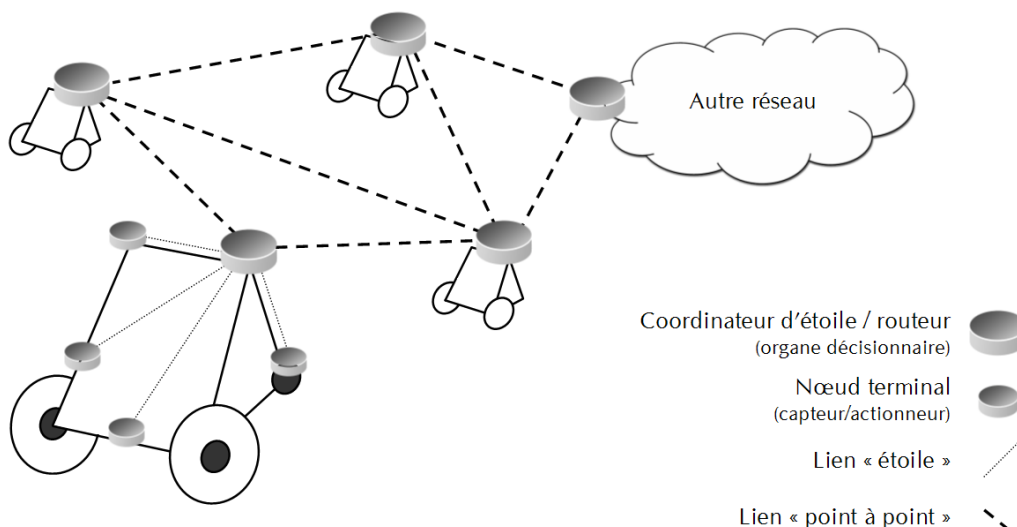


FIG. 3.2 – Topologie proposée pour la communication entre robots

Les deux topologies évoquées doivent, bien entendu, pouvoir coexister et les mécanismes de QoS implémentés pour chacune des méthodes d'accès ne doivent pas se perturber, mais au contraire garantir leurs propriétés respectives sur l'ensemble du réseau.

### 1.2.5 Débit nécessaire

A première vue, le débit nécessaire pour notre application est assez faible car les informations issues des capteurs sont généralement peu volumineuses, tout au plus quelques octets à chaque transfert suscité

par une nouvelle mesure. Cependant, le débit nécessaire peut rapidement augmenter :

- *si les capteurs envoient des informations très fréquemment* ; un capteur très sollicité ou un actionneur nécessitant d'être informé en permanence risque de monopoliser le médium et augmenter considérablement les besoins en débit,
- *si le nombre de capteurs est élevé* ; le médium étant partagé entre toutes les entités communicantes, plus il y a d'entités, plus le débit sur le médium doit être important pour satisfaire toutes les communications,
- *si le nombre de robots est important* ; même remarque que la précédente, les robots partageant le même médium,
- *si le protocole utilisé génère beaucoup d'informations non utiles (en-tête, etc.)* ; typiquement, dans notre application, les données échangées sont très courtes mais peuvent être nombreuses et fréquentes. Si la technologie réseau choisie présente un *overhead*<sup>◇</sup> important ou nécessite l'échange de nombreux messages de gestion ou de signalisation, le débit nécessaire sera important, même si la charge utile est faible,
- *si l'accès au médium est mal organisé* ; l'accès étant multiplexé et les informations véhiculées courtes (trames courtes), les occasions d'accès au médium seront très fréquentes. Si l'accès au médium est mal coordonné, le phénomène des collisions risque de générer de nombreux échecs de transmission et, de ce fait, augmenter les retransmissions, très consommatrices en bande passante (et accessoirement en énergie).

Il est donc nécessaire de dimensionner le réseau en conséquence (nombre de robots et nombre de capteurs par robot limité comme nous l'avons évoqué plus haut) et de choisir une technologie de transmission présentant à la fois un protocole simple et peu consommateur de bande passante (ce qui rejoint les attentes d'un point de vue énergétique) et une méthode d'accès au médium très performante et adaptée aux « petits » paquets d'informations envoyés régulièrement. Les analyses de performances présentées dans le chapitre 4 indiqueront des valeurs numériques de référence, acceptables en terme de latence et de débit.

### 1.2.6 Conclusion

L'application typique présentée ici nécessite finalement un système de communication s'apparentant à un *Réseau Local Industriel sans fil*. Les besoins de communication peuvent être résumés en quelques points ; le système proposé doit :

1. préférer la fiabilité et la stabilité à un débit important,
2. mettre en œuvre un protocole léger,
3. proposer des outils pour garantir la remise des données au destinataire,
4. permettre de prévoir plusieurs classes de trafics (utilisation du médium, notion de priorités),
5. connaître a priori les nœuds terminaux potentiels,
6. être peu consommateur d'énergie.

Dans le chapitre précédent, nous avons présenté dans le détail la norme IEEE 802.15.4/ZigBee qui propose des couches basses très intéressantes au vu de ce que requiert notre application. Nous nous sommes donc tournés vers cette technologie que nous avons largement modifiée pour pouvoir proposer une nouvelle méthode d'accès au médium totalement déterministe. De plus, la disponibilité matérielle, le coût très abordable de cette technologie et la possibilité de pouvoir reprogrammer toute la couche 2 nous ont permis d'entrevoir des possibilités de prototypage réel très intéressantes dont les résultats seront exposés dans le chapitre 4.

## 2 La proposition protocolaire

Nous venons d'évoquer les besoins en communication d'une application typique de robotique mobile et dans le chapitre précédent, nous avons présenté les normes et spécifications IEEE 802.15.4 et ZigBee ; nous avons alors souligné que le mécanisme de réservation de bande passante (GTS<sup>◇</sup>) proposé par l'IEEE est intéressant, mais pourrait être amélioré par la proposition d'une méthode d'accès au médium plus orientée « déterminisme » et par l'introduction d'une certaine souplesse au niveau des possibilités de réservation du médium sans fil. Dans cette partie, nous présentons une méthode d'accès au médium totalement déterministe [VDB 06] [VDB 07] qui permettrait à IEEE 802.15.4/ZigBee [IEE2 03] [ZBAL 05] ou à d'autres technologies WLAN/WPAN de pouvoir être utilisées dans le cadre d'applications à fortes contraintes temporelles comme l'application typique qui vient d'être évoquée.

### 2.1 Nouvelles fonctionnalités proposées

Avant de détailler les mécanismes que nous proposons, listons ci-après les fonctionnalités que nous ajoutons par rapport à l'existant :

1. Dans 802.15.4, seuls les nœuds terminaux peuvent demander un GTS pour eux-mêmes. Dans notre proposition, **un coordinateur peut décider d'attribuer un GTS à n'importe lequel de ses correspondants, même si celui-ci n'est pas encore associé à ce même coordinateur.** Cette nouvelle fonctionnalité va permettre de réaliser des **associations déterministes** au réseau. Nous appellerons par la suite ce GTS attribué au préalable  $PDS^{\diamond}$ , pour *Previously Dedicated Slot*.
2. Dans 802.15.4, aucun mécanisme de communication entre coordinateurs n'est prévu pour garantir que deux GTS ne soient pas attribués au même instant par deux coordinateurs différents qui cohabitent dans la même zone de portée radio et sur le même canal radio. Dans notre proposition, **nous proposons un mécanisme pour empêcher les « collisions de GTS »** entre étoiles différentes.
3. Dans 802.15.4, les GTS sont placés dans la CFP<sup>◇</sup> qui se trouve toujours en fin de supertrame. Dans notre proposition, **il n'y a plus de délimitation pour la CFP, elle est répartie, diffuse, dans la supertrame.** Les GTS peuvent être disposés à tout instant dans la supertrame, ce qui va nous permettre d'optimiser cette répartition en tenant compte des GTS attribués dans les autres étoiles. Comme dans 802.15.4, cette répartition est indiquée à tous les nœuds terminaux par le coordinateur dans la trame *beacon* qu'il envoie systématiquement en début de supertrame.
4. Dans 802.15.4, les GTS alloués sont présents dans chaque supertrame. Dans notre proposition, **chaque GTS peut être présent, au choix, dans toutes les supertrames, dans une supertrame sur deux, une sur quatre ou une sur huit, etc.** On parlera de plusieurs niveaux de réservation notés  $n$ , avec, pour le cas de notre étude<sup>3</sup> :

$$n = 0, 1, 2 \text{ ou } 3.$$

La période  $P$  des supertrames comprenant ce GTS d'un niveau de réservation  $n$  est définie ainsi :

$$P = 2^n$$

Un niveau de réservation  $n = 0$  réservera un slot dans chaque supertrame (fréquence la plus élevée), un niveau  $n = 1$  réservera un slot toutes les deux supertrames, un niveau  $n = 2$  toutes les quatre supertrames, un niveau  $n = 3$  toutes les huit supertrames (fréquence la plus faible). Ce dernier point va permettre à des trafics différents mais à  $QoS^{\diamond}$  de pouvoir cohabiter au sein d'une même étoile. En revanche, comme dans la norme initiale, les GTS peuvent toujours occuper plusieurs slots de la même supertrame si nécessaire.

---

<sup>3</sup>Pour des questions de restriction au niveau de l'implémentation, nous avons fixé arbitrairement une limite maximum de 8 supertrames, soit quatre niveaux  $n$ , mais cette limite peut être tout à fait repoussée.

5. Dans 802.15.4, les GTS sont attribués pour un nombre limité de supertrames, la prolongation du bail ne pouvant se faire que par une nouvelle demande, c'est à dire une reprise complète du processus de requête. De plus, comme cette demande est effectuée par l'utilisation du protocole d'accès au médium non déterministe CSMA/CA<sup>◇</sup>, la continuité du service déterministe ne peut être garantie. Dans notre proposition, **les GTS sont alloués sans limite temporelle, jusqu'à la réception d'une demande de relâchement venant du nœud terminal concerné ou d'une notification de relâchement venant du coordinateur** (*timeout* sur inactivité, par exemple).
6. Dans 802.15.4, toutes les entités du réseau émettent à une puissance constante dans le temps. Cette puissance peut être différente d'un élément à l'autre, mais la norme ne prévoit pas de variation par le protocole. Dans la proposition faite, **les puissances d'émission peuvent varier dynamiquement pour créer artificiellement des zones sans collision**, c'est-à-dire des zones géographiques distinctes et identifiées comme suffisamment éloignées pour que deux nœuds terminaux puissent accéder au médium simultanément sans que la réception de leurs messages respectifs ne soit altérée. Grâce à cette fonctionnalité, deux coordinateurs identifiés comme « suffisamment loin l'un de l'autre » peuvent attribuer deux GTS sur le même slot temporel.

Ces nouvelles fonctionnalités étant exposées, nous allons détailler leur mise en oeuvre dans la suite de ce chapitre.

## 2.2 Problématique de la cohabitation de plusieurs étoiles sur un médium commun

Dans le cadre d'un réseau constitué d'une seule étoile, le coordinateur unique est le seul à décider de l'attribution ou non des GTS. Si plusieurs coordinateurs cohabitent dans la même zone et occupent le même canal, les décisions d'attribution de GTS doivent être prises collégalement. De ce fait, il n'est pas possible de considérer un réseau à plusieurs étoiles comme une simple agrégation d'étoiles gérées individuellement. Dans l'état actuel de 802.15.4, un coordinateur annonce la répartition des GTS dans sa trame balise : une solution simple consisterait en une écoute passive pour chaque coordinateur des beacons émis par ses voisins pour éviter d'attribuer un GTS à l'instant où un coordinateur voisin a déjà attribué un GTS à l'un de ses nœuds terminaux.

De plus, dans notre proposition, nous prévoyons la possibilité pour certains GTS de revenir cycliquement toutes les deux, quatre ou huit supertrames. Comme pour l'attribution d'un GTS classique, cette décision doit prendre en compte la charge du réseau, mais notre proposition implique une vision globale à plus long terme. En effet, plus les GTS sont attribués avec rareté, plus le volume d'informations à prendre en compte pour le mécanisme de réservation collégial est important : plus le niveau de réservation  $n$  est élevé, plus la portée de la vision doit être grande. Cette fonctionnalité augmente la complexité relative de la prise de décision collégiale pour l'attribution des GTS car la trame balise n'annonce que la répartition des GTS pour la supertrame qui débute, et non pour les suivantes. Les coordinateurs voisins ne peuvent pas connaître l'intégralité des décisions prises en interne du coordinateur, puisqu'ils ne connaissent, au moment de la réception de la balise, que la répartition pour la supertrame qui vient de commencer. Il est donc nécessaire de mettre en place une solution qui empêchera plusieurs coordinateurs d'attribuer un même slot temporel à plusieurs nœuds du réseau.

Deux solutions sont envisageables pour palier à ce problème. **Une première solution** consiste pour chaque coordinateur à diffuser toute décision prise à l'ensemble des voisins, par exemple dans les beacons, que cette décision concerne la supertrame en cours ou les suivantes. Ainsi, les voisins sont informés et peuvent éviter d'attribuer des slots réservés par d'autres coordinateurs. Cette solution a l'avantage d'être décentralisée, mais ne règle cependant pas deux problèmes :

1. tous les coordinateurs doivent être à portée radio les uns des autres,

2. dès leur arrivée dans le réseau, les coordinateurs doivent répartir leurs trames beacons respectives de manière déterministe pour ne pas créer de collisions ni avec les autres trafics établis sur le réseau ni avec les autres beacons, ce qui est un problème complexe, surtout pour les phases transitoires d'insertion et de départ d'un coordinateur dans le réseau mobile.

**Une seconde solution, qui est celle qui a été retenue**, consiste à centraliser toutes les demandes de réservation du médium. Les demandes sont acheminées jusqu'à une entité spéciale qui a connaissance de tous les slots réservés à plus ou moins brève échéance : cette entité donne son accord et nous évitons alors les collisions de GTS. Si cette station à rôle central est à portée radio de tous les coordinateurs, elle peut du même coup régler les problèmes du coordinateur caché et de la synchronisation entre coordinateurs, en répartissant elle-même les beacons de tous les coordinateurs. Cette répartition peut se faire très simplement, sous la forme de plusieurs GTS qu'elle accorde aux coordinateurs (un GTS par coordinateur) et qui serviront à émettre les beacons de ceux-ci sans risquer de provoquer des collisions. Comme nous l'avons vu dans le chapitre 2, ZigBee prévoit dans sa spécification une entité spéciale, le *coordinateur de PAN* (ou *supercoordinateur*) dont le rôle central est d'attribuer des adresses aux nœuds qui s'insèrent dans le réseau, d'opérer un changement de canal si besoin, etc. Nous avons donc ici retenu cette deuxième solution, qui est certes plus centralisée mais qui « correspond mieux » à l'idée de base de ZigBee, qui est un réseau proposant une architecture assez centralisée. Le supercoordinateur sera donc en mesure de distribuer des GTS dédiés aux beacons des coordinateurs d'étoile. Nous appellerons par la suite ces GTS dédiés aux beacons  $GBS^\diamond$ , pour *Guaranteed Beacons Slots*.

## 2.3 Étude du protocole mis en œuvre

### 2.3.1 Éléments du réseau, liens et portées

Comme le proposent 802.15.4 et ZigBee, notre proposition prévoit trois catégories de nœuds communicants :

1. un supercoordinateur, unique,
2. un coordinateur pour chaque étoile,
3. un ou plusieurs nœuds terminaux par étoile.

Dans un premier temps, nous considérerons que le supercoordinateur ne fait que « supercoordonner », bien qu'il puisse lui aussi gérer sa propre étoile et ses nœuds terminaux comme un coordinateur normal.

Le réseau compte deux types de liens de communication :

1. un premier type entre le supercoordinateur et ses coordinateurs,
2. un second type entre les coordinateurs et leurs nœuds terminaux, dialogues réalisés au sein de chaque étoile.

La topologie proposée implique des portées minimales qui sont définies ainsi :

1. Les coordinateurs d'étoiles doivent être à portée du supercoordinateur,
2. Les nœuds terminaux doivent être à portée de leur coordinateur,

La première hypothèse règle le problème du coordinateur caché mais contraint fortement le système de communication radio. Nous serons certainement amenés à proposer ultérieurement une amélioration du protocole pour parvenir à une solution moins contraignante<sup>4</sup>.

---

<sup>4</sup>Nous avons vu dans le chapitre 2 que certains modules ZigBee comme les XBEEPRO de MAXSTREAM [MAXS] permettent d'atteindre des portées très importantes (1,6 km en champ libre) ; cette hypothèse n'est donc pas si contraignante dans la réalité d'une flotte de robots coopérants éloignés tout au plus de quelques dizaines de mètres...

La topologie proposée et les portées minimales sont représentées sur la figure 3.3. Bien entendu, les portées radio peuvent être plus étendues; ici, seules les portées minimales et nécessaires au bon fonctionnement du protocole ont été représentées.

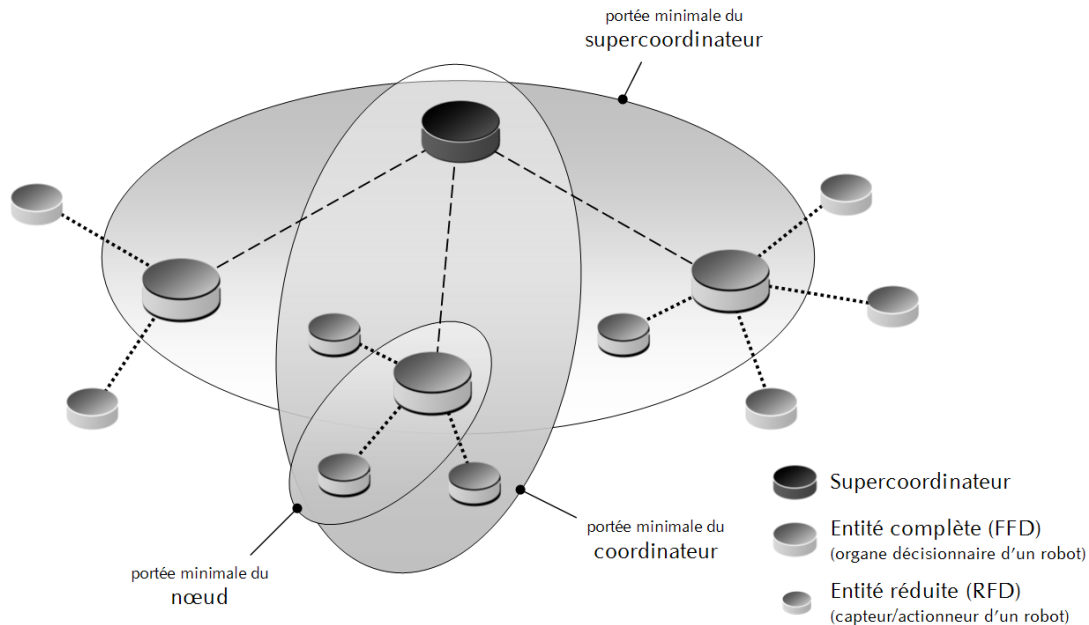


FIG. 3.3 – Éléments du réseau, liens et portées minimales pour le bon fonctionnement du protocole

### 2.3.2 Organisation temporelle de l'accès au médium

Dans notre proposition, nous conservons la notion de supertrame. Celle-ci permet de numérotter les slots et permet une synchronisation de toutes les entités de notre réseau. De plus, le mécanisme de réservation du médium – même s'il comporte quelques failles que nous nous proposons de combler – s'appuie avantageusement sur le principe de supertrame.

Dans notre proposition, la supertrame compte toujours 16 slots et commence toujours par une trame balise qui occupe le premier slot. En revanche, dans 802.15.4, chaque coordinateur débute sa supertrame dès qu'il le souhaite; chaque coordinateur étant indépendant sur le plan de la synchronisation, il n'y a aucune raison qu'elles soient synchrones : des *collisions de beacons* sont donc probables (cf. chapitre 2, section 2.4.6, page 76). Dans notre proposition, seul le supercoordonateur est habilité à choisir le moment où la supertrame débute. De plus, il n'y a plus qu'une supertrame pour tout le réseau car tous les coordinateurs sont synchronisés.

Nous appelons *superbeacon* le Beacon émis par le supercoordonateur; le superbeacon occupe systématiquement le slot 0 de chaque supertrame. Les beacons « normaux » (ceux émis par les coordinateurs d'étoile) sont placés par le supercoordonateur dans les GBS. Pour les nœuds terminaux d'une étoile, la supertrame ne commence pas sur le slot 0 pour se finir sur le slot 15, mais commence sur le slot  $i$  pour se terminer sur le slot  $i - 1$ . La figure 3.4 illustre une répartition des beacons via les GBS pour plusieurs coordinateurs dans une étoile. Ici, la supertrame du coordinateur 1 commence au slot 4 et se termine au slot 3 suivant. Le fait de « mélanger » les supertrames de chaque coordinateur dans la supertrame du supercoordonateur (au lieu de les faire se succéder dans le temps) permet aux nœuds terminaux de pouvoir émettre des messages à des intervalles de temps plus rapprochés, favorisant ainsi les transports d'informations qui nécessitent une faible latence de transmission.

Pour que les beacons puissent être diffusés sans risquer d'entrer en collision avec un autre message, les slots qu'ils occupent leur sont dédiés (GTS montants) par le supercoordonateur comme nous l'avions

évoqué dans la partie 2.2. C'est le supercoordonateur qui choisit la valeur de  $i$  pour chaque coordonnateur. Ainsi, comme nous l'avons proposé dans la section précédente, le supercoordonateur peut répartir les émissions des beacons dans la supertrame, par exemple en la divisant en parts égales comme dans le cas de la figure 3.4. Dans ce cas illustré,  $i_{(1)} = 4$ ,  $i_{(2)} = 8$  et  $i_{(3)} = 12$  (1, 2 et 3 étant les numéros de 3 coordonnateurs présents dans le réseau).

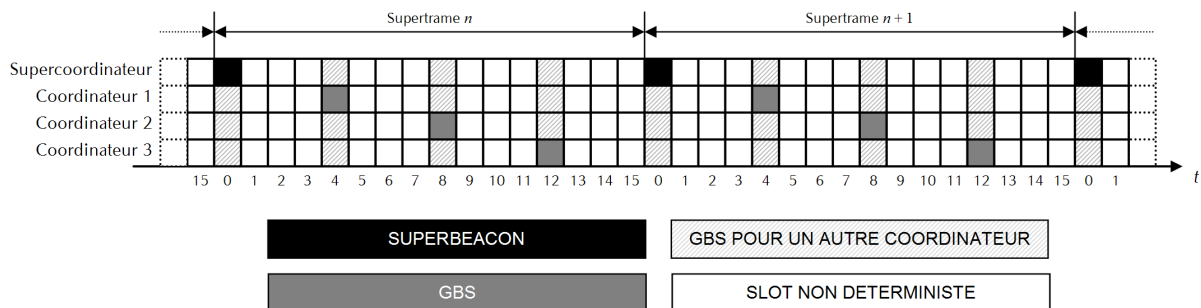


FIG. 3.4 – Répartition des beacons au sein de la supertrame

Le fait de centraliser la répartition des beacons dans le supercoordonateur peut en même temps régler le problème du coordonnateur caché ainsi que l'entrée déterministe d'un coordonnateur dans le réseau, comme nous allons le voir plus bas dans le paragraphe 2.3.4.

### 2.3.3 Mécanisme de demande de réservation du médium

Pour pouvoir accorder un GTS à l'un de ses nœuds terminaux, un coordonnateur doit d'abord envoyer une requête au supercoordonateur et obtenir une réponse positive. Seul le supercoordonateur a une vision exhaustive de la répartition du médium pour les  $2^{n_{MAX}}$  supertrames à venir (dans notre cas,  $2^{n_{MAX}}$  vaut 8). Compte tenu de nos objectifs en terme de déterminisme pour l'accès au médium, la requête de demande de réservation envoyée par un coordonnateur au supercoordonateur doit être également émise sur le médium de manière déterministe (toute la chaîne doit être déterministe pour que le résultat soit déterministe!). Pour ce faire, les dialogues entre le supercoordonateur et les coordonnateurs prennent place dans des slots déjà attribués par le supercoordonateur pour chacun de ses coordonnateurs : les GBS. La réutilisation des slots dédiés aux beacons permet d'économiser l'occupation du médium, il n'est pas nécessaire d'allouer un slot supplémentaire aux coordonnateurs pour dialoguer avec le supercoordonateur puisqu'ils possèdent déjà un tel slot. Dans notre proposition, les messages **GTS.requests** seront inclus dans le champ data de la trame Beacon au lieu d'être émis via une trame MAC à part. La spécification de ce procédé d'encapsulation sera décrite dans la partie 2.4 de ce chapitre.

### 2.3.4 Allocations au préalable : arrivées déterministes dans le réseau

Nos objectifs en terme d'accès déterministe au médium impliquent que toutes les opérations réalisées sur le réseau soient faites de manière déterministe. Notre proposition prévoit donc une possibilité « de réservation de slots au préalable », pour certains nœuds, c'est-à-dire avant même que ces nœuds soient associés au réseau ou qu'ils aient demandé des ressources. Cette fonctionnalité est indispensable pour garantir les propriétés déterministes de notre réseau pendant toutes les phases de son fonctionnement. Elle est rendue possible car le réseau considéré est « petit » et il est possible de connaître toutes les entités potentielles et de leur affecter manuellement et a priori une adresse avant la phase de création de ce WPAN.

Grâce à cette fonctionnalité, le supercoordonateur peut, dès sa mise en fonction (c'est-à-dire dès le début de la création du réseau), réserver quelques slots pour certains nœuds, par exemple pour les coordonnateurs attendus ou des nœuds critiques. Cette réservation se concrétise par l'attribution de GBS



ou GTS montants à un nœud avant son arrivée dans le réseau. Cette attribution au préalable peut se faire avec un niveau  $n_{MAX}$  (la fréquence de réservation la plus faible, un GBS/GTS toutes les huit supertrames) ce qui laisse la possibilité au nœud concerné de pouvoir émettre de façon déterministe une trame d'association quelque temps après son réveil, tout en minimisant les pertes de bande passante dans l'attente de l'arrivée de ce nœud. A titre d'exemple, pour  $BO^\diamond = 3$ , c'est-à-dire un *intervalle inter-beacon* de 122,88 ms, une réservation au préalable avec un niveau  $n = 3$  permet de pratiquer l'arrivée déterministe d'un certain nœud toutes les 983,04 ms (soit environ toutes les secondes<sup>5</sup>) pour seulement  $1/128^{ime}$  (soit 0,78%) de la bande passante perdue, ce qui est relativement faible.

Le graphique de la figure 3.5 donne les périodes de cycles déterministes et la bande passante perdue en fonction du niveau de réservation  $n_{PDS}$  (0, 1, 2 ou 3) et pour différentes valeurs de  $BO$ . Notons que la part de bande passante perdue ne dépend pas du paramètre  $BO$ .

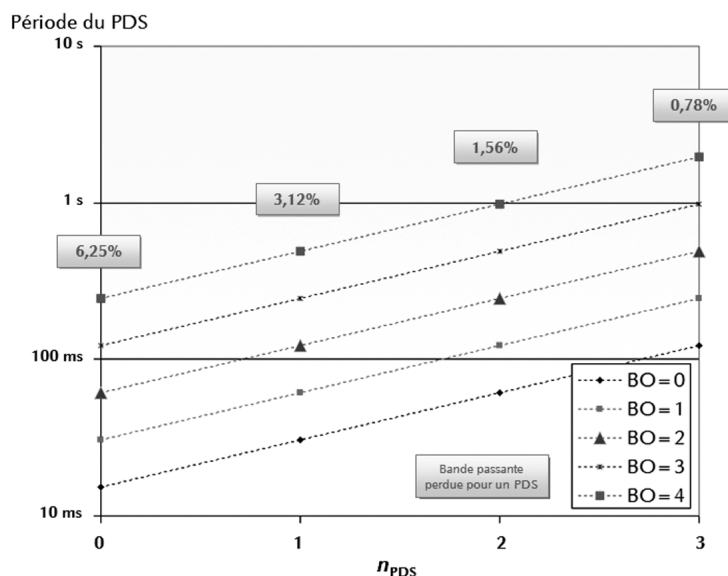


FIG. 3.5 – Périodes des cycles déterministes en fonction du niveau de réservation du PDS

Le graphique de la figure 3.5 constitue un abaque intéressant pour l'utilisateur désirant dimensionner le réseau en fonction d'un cahier des charges imposant une certaine latence de transmission.

### 2.3.5 Gestion des acquittements

Dans notre proposition, la gestion des acquittements reste similaire à ce que propose le standard IEEE 802.15.4. L'émetteur d'une trame positionne ou non le bit **ACK\_request** dans chaque trame émise. A la réception, si le bit est positionné, le récepteur doit acquitter en envoyant une trame d'acquittement. Le temps inter-trames séparant la trame à acquitter et l'acquittement n'est pas le même dans toutes les situations. Deux cas se présentent :

1. si la trame est envoyée dans le cadre d'un accès au médium déterministe, la transmission de l'acquittement doit se faire après un temps de *aTurnaroundTime* symboles (12 symboles, soit 192  $\mu$ s) après la réception du dernier symbole reçu – comme le définit le standard 802.15.4 pour les trames envoyées dans un GTS,
2. si la trame est envoyée dans le cadre d'un accès au médium aléatoire, la transmission de l'acquittement doit se faire au début du slot CSMA/CA suivant (là encore, comme 802.15.4 le prévoit).

<sup>5</sup>Bien entendu, rien n'empêche ce nœud de tenter une association le plus tôt possible en suivant la méthode classique en mode *best-effort*, comme la norme actuelle le permet.

Dans tous les cas, l'émetteur considère qu'une trame non acquittée en moins de *macAckWaitDuration* symboles (54 symboles, soit 864  $\mu$ s) constitue un échec et doit être retransmise – l'émetteur a droit à *MaxFrameRetries* (3) tentatives. Comme nous l'avons vu dans le chapitre précédent (chapitre 2, section 2.3.3.4, page 68), la trame d'acquiescement est très simple (en-tête MAC et numéro de séquence de la trame acquittée) et ne permet d'acquiescer qu'une seule trame à la fois (*Send and wait*). De plus, seules les trames unicast de type Données et Commandes MAC peuvent être acquittées.

En revanche, à la différence de ce que propose le standard 802.15.4 et afin minimiser les accès déterministes au médium, notamment entre entités de niveaux différents, nous proposons d'encapsuler certains messages de gestion MAC dans les trames balises. De ce fait, ces messages encapsulés ne sont plus acquittés et, s'ils sont perdus, entraînent une latence dans le traitement de la requête qu'ils portent car il ne peuvent être retransmis immédiatement<sup>6</sup>. Cette latence introduite sera l'objet d'une étude de performances dans le chapitre suivant.

### 2.3.6 Gestion et diffusion des informations de réservation

Nous l'avons vu plus haut, nous avons choisi une architecture centralisée où le supercoordonateur est le seul nœud habilité à attribuer des PDS/GBS/GTS. De ce fait, tous les messages de demande de réservation de médium (*GTS.request*) remontent jusqu'à lui.

Dans le cadre strict de 802.15.4, non seulement l'attribution et la gestion des GTS sont deux tâches locales à l'étoile mais les GTS, s'il y en a, sont présents dans chaque supertrame. De ce fait, la réponse à la demande *GTS.request* est contenue dans le champ des descripteurs de GTS (*GTS descriptor*) qui se trouve dans chaque trame beacon :

- Si un descripteur de GTS portant l'adresse du nœud demandeur est présent, le GTS a bien été attribué : la requête a donc été honorée,
- Si le beacon ne contient aucun descripteur de GTS avec l'adresse du nœud demandeur, le GTS n'a pu être attribué : la requête n'a donc pas été honorée.

Dans la mesure où dans notre proposition, les GTS restent périodiques mais ne se retrouvent pas forcément dans toutes les supertrames ( $n_{GTS} > 0$ ), la réponse à une demande de GTS sous forme d'un descripteur de GTS dans le superbeacon qui suit la demande ne peut être satisfaisante car si le GTS est effectivement attribué, mais « bien plus tard » (c'est-à-dire si  $n_{GTS}$  est grand), le nœud demandeur n'obtiendra sa réponse qu'au début de la supertrame contenant son premier GTS, ce qui pose un gros problème de latence, surtout pour les nœuds critiques. Il serait préférable que la réponse soit immédiate, qu'elle soit positive ou négative.

Nous proposons donc de spécifier un nouveau message (*GTS.response*) qui pourra être envoyé immédiatement par le supercoordonateur et confirmer si le GTS a été ou non attribué, et avec quel niveau de réservation  $n$ . De plus, dans la mesure où, dans notre proposition, les demandes de GTS sont relayées par les coordinateurs jusqu'au supercoordonateur, la présence d'un message *GTS.response* facilitera la communication de bout en bout du réseau. La structure de ce message sera spécifiée plus loin dans la partie 2.4 de ce chapitre.

### 2.3.7 Un exemple de demande de réservation du médium

Dans cette partie, nous proposons d'illustrer le déroulement temporel du dialogue protocolaire nécessaire pour l'association déterministe d'un nœud terminal au réseau. L'exemple porte sur un réseau dont la topologie est représentée figure 3.6. Ce réseau est minimaliste, il ne comprend qu'un nœud terminal, son coordinateur et le supercoordonateur du réseau. Le déroulement temporel du dialogue protocolaire

---

<sup>6</sup>Précisons ici qu'un message de gestion MAC se voit acquitté simplement pour confirmer qu'il a été bien reçu par le destinataire. En aucun cas, l'acquiescement constitue une réponse favorable à la requête effectuée.

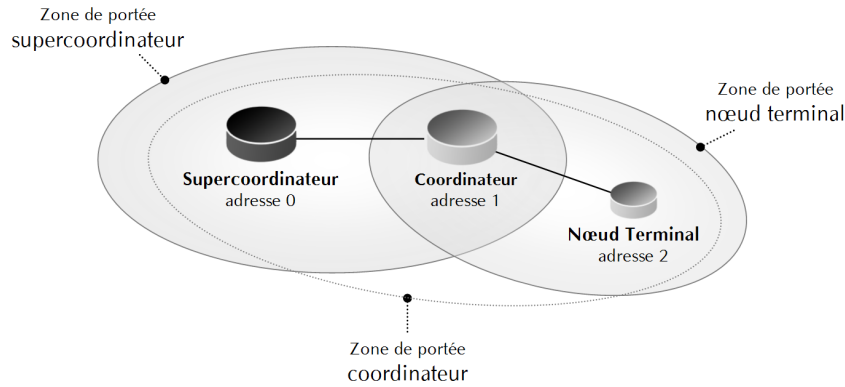


FIG. 3.6 – Topologie du réseau mis en place pour l'exemple

est illustré par la figure 3.7. L'objectif ici est d'illustrer l'échange des messages nécessaire pour qu'un nœud terminal obtienne un GTS, et ce, de manière totalement déterministe, y compris dans sa phase de connexion (association) au réseau.

À l'état initial, les trois éléments du réseau se « connaissent », c'est à dire que des adresses ont été attribuées afin de permettre aux différents nœuds de pouvoir réaliser une entrée déterministe sur le réseau. Dans cet exemple, le supercoordonneur a l'adresse 0, le coordinateur l'adresse 1 et le nœud terminal l'adresse 2. Le supercoordonneur et le coordinateur sont synchronisés, ils s'échangent des beacons régulièrement, à chaque supertrame ( $n_{GBS} = 0$ ). En revanche, bien qu'il soit connu de son coordinateur et du supercoordonneur, le nœud terminal n'est pas encore réveillé. Cependant, l'applicatif présent dans le supercoordonneur sait que ce nœud terminal est lié à une fonction critique ; il lui a donc attribué un PDS qui revient, cycliquement, avec un niveau 3 ( $n_{PDS} = 3$ , donc toutes les 8 supertrames).

Sur la figure 3.7, les flèches représentent les messages échangés entre un nœud source et un nœud destination (pointe de la flèche). Dans la marge de droite, les messages et leur contenu (entre parenthèses) sont spécifiés dans un formalisme dont la syntaxe est précisée ci dessous. Cette syntaxe varie d'un type de message à l'autre. Dans l'exemple illustré, nous allons rencontrer quatre sortes de messages :

1. Les demandes de PDS/GBS/GTS : elles sont envoyées par un nœud terminal ou un coordinateur et envoyées à leur entité *père*. Elles contiennent simplement le niveau de ressource demandé  $n$ . Par exemple, le message `GTS_req(level 0)` indiquera que le message est une demande de PDS/GBS/GTS et que l'initiateur de ce message souhaiterait obtenir un PDS/GBS/GTS de niveau  $n = 0$ . Ces messages sont relayés par les coordinateurs en étant encapsulés dans leurs beacons (voir plus bas).
2. Les réponses aux demandes de PDS/GBS/GTS : elles sont envoyées en réponse à un `GTS_req()` par le supercoordonneur et relayées par les coordinateurs. Elles indiquent le niveau du PDS/GBS/GTS attribué. Par exemple, le message `GTS_resp(level 0)` indique d'un PDS/GBS/GTS de niveau 0 a été attribué<sup>7</sup>. Les réponses aux demandes de PDS/GBS/GTS sont systématiquement encapsulées dans le superbeacon ou le beacon.
3. Les beacons – ou superbeacon (même syntaxe) : ils contiennent la liste des PDS/GBS/GTS attribués pour la supertrame en cours (*GTS descriptors*). Dans l'exemple illustré, un PDS/GBS/GTS attribué est noté ainsi : `<slot>:<adresse>`. Si le beacon doit annoncer l'attribution de plusieurs PDS/GBS/GTS, ceux-ci sont alors séparés par des virgules. Par exemple, le message `Superbeacon (8:1,6:2)` indiquera que le message transmis est de type *beacon* et que deux PDS/GBS/GTS ont été attribués pour la supertrame qui débute ; le premier PDS/GBS/GTS se trouve sur le slot n°8 et est dédié au nœud d'adresse 1, le second PDS/GBS/GTS se trouve sur le slot n°6 et est

<sup>7</sup>Le PDS/GBS/GTS est attribué au destinataire du message, le message étant relayé.

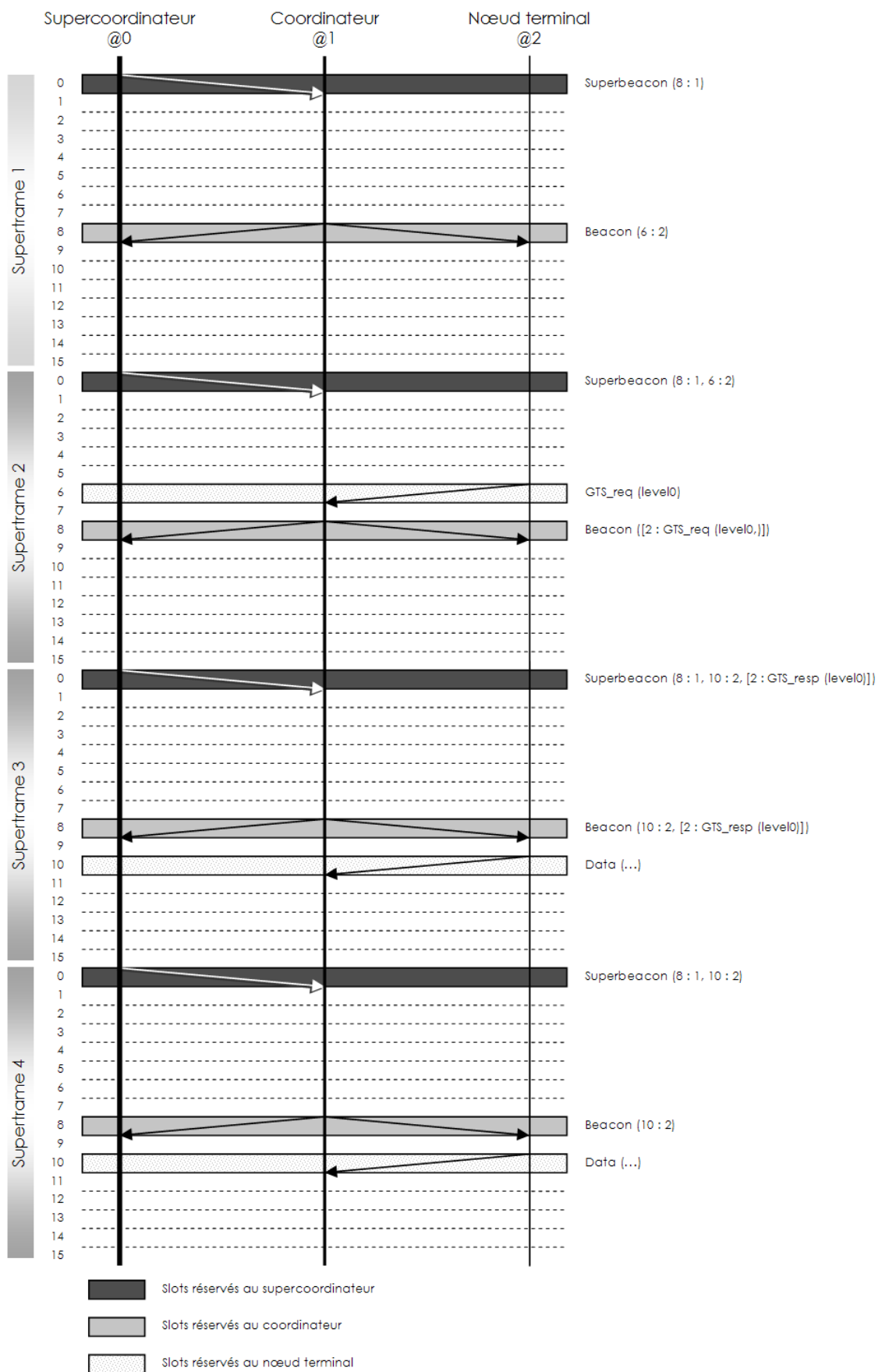


FIG. 3.7 – Un exemple de déroulement du protocole

dédié au nœud d'adresse 2. Comme on l'a vu plus haut, les beacons peuvent également encapsuler d'autres messages dans le cadre d'un relai de trame ; dans ce cas, le message relayé est indiqué entre crochets et l'adresse du destinataire est indiquée dans les crochets : par exemple, le message `Beacon(10:2, [2:GTS_resp(level 0)])` indique que le slot n°10 est dédié au nœud d'adresse 2, mais aussi qu'un message de type « réponse à la demande d'un PDS/GBS/GTS » à destination du nœud 2 doit être relayé par le coordinateur correspondant.

4. Les données : les données ne comportent pas de paramètres spéciaux.

Pendant tout le dialogue, le supercoordinateur diffusera les superbeacons sur tous les slots 0 alors que le coordinateur diffusera ses beacons sur tous les slots 8 ( $n_{GBS} = 0$ ) ; on retrouve cette répartition dans la première supertrame de la figure, où les beacons constituent l'unique trafic pendant cette première période. Le coordinateur sait que son GBS se trouve au slot 8 car le superbeacon l'indique par le message `Superbeacon(8:1)`. Pour répondre à nos objectifs en terme de déterminisme, c'est-à-dire pour pouvoir pratiquer une demande de ressource déterministe, un PDS de niveau ( $n_{PDS} = 3$ ) a donc été alloué au nœud terminal, selon le principe de l'allocation au préalable vu plus haut. Ce slot réservé revient donc toutes les huit supertrames ( $n_{PDS} = 3$ ) et grâce à ce slot dédié, le nœud terminal peut prendre possession du médium de manière déterministe, par exemple pour dire qu'il est maintenant réveillé et éventuellement demander un ou plusieurs GTS.

Au début de notre exemple, le nœud terminal vient de se réveiller. Il reste en réception pour rechercher le beacon de son coordinateur ; une fois le beacon identifié et interprété, le nœud terminal est synchronisé et peut passer en mode « suivi de beacon » (réception du beacon, interprétation, puis veille) jusqu'à recevoir un beacon lui indiquant la présence de son PDS. Son  $n_{PDS}$  étant de 3, il attend au plus 8 supertrames<sup>8</sup>. Le PDS avait été positionné bien auparavant par le supercoordinateur : puisqu'il n'a pas reçu de message de désallocation du PDS par le supercoordinateur, le coordinateur du nœud terminal sait que le PDS se trouvera au cours de la seconde supertrame de la figure. Ce PDS est donc annoncé dans le beacon du coordinateur, via le message `Beacon(6:2)`. Le nœud terminal étant en mode suivi de beacon, il apprend alors qu'il pourra bénéficier d'un accès déterministe au prochain slot n°6.

Quand arrive le slot n°6, le nœud terminal émet une demande de ressources de niveau 0 par le message `GTS_req(level 0)`. Ce message est reçu par le coordinateur qui le relaie au supercoordinateur dans le beacon suivant (procédé d'encapsulation du message dans le beacon). Le supercoordinateur reçoit la requête, la traite et retourne une réponse dans le superbeacon suivant `Superbeacon(8:1, 10:2, [2:GTS_resp(level 0)])`. A son tour, le coordinateur relaie cette réponse au nœud terminal dans le beacon qui suit via le message `Beacon(10:2, [2:GTS_resp(level 0)])`. Le nœud terminal prend alors connaissance de la réponse et peut (ou non, si la réponse est négative) utiliser les ressources qu'il a demandées. Dans le cas de l'exemple, la réponse est positive et le nœud terminal obtient un GTS de niveau 0 (présence dans toutes les supertrames à venir).

### 2.3.8 Une politique d'accès déterministe par défaut

Dans notre topologie, nous avons supposé que d'une part, les coordinateurs doivent tous être à portée radio du supercoordinateur et d'autre part, que les nœuds terminaux doivent être à portée radio de leur coordinateur.

En revanche, les coordinateurs ne sont pas obligés d'être à portée les uns des autres<sup>9</sup> et les nœuds terminaux ne sont pas obligés d'être à l'écoute du supercoordinateur. Cette dernière hypothèse concernant les nœuds terminaux se justifie amplement d'un point de vue énergétique : conformément à ce qui est proposé dans la norme 802.15.4, le nœud simple, contraint énergétiquement, suit un cycle de fonctionnement adapté qui limite sa consommation énergétique : il se réveille, écoute le beacon de son

<sup>8</sup>Cas le plus défavorable : en effet, il peut aussi le trouver immédiatement ; le temps nécessaire dépend de l'instant de réveil !

<sup>9</sup>Nous avons en effet vu plus haut que le problème du coordinateur caché est réglé par le supercoordinateur qui rappelle dans tous ses beacons la répartition temporelle qui est faite dans la supertrame qui commence.

coordinateur, utilise le médium si nécessaire, puis se rendort. D'un point de vue énergétique, un double suivi de beacon (beacon et superbeacon) serait très mauvais et entraînerait des pertes énergétiques non négligeables.

Les considérations qui viennent d'être exposées soulèvent un problème concernant le temps de diffusion des informations sur la répartition temporelle du médium. La nécessité de relayer l'information dans la structure à trois niveaux (supercoordinateur, coordinateur, nœud terminal) et la répartition dans le temps des beacons au sein de la supertrame entraînent un temps de propagation non négligeable avant que tous les nœuds du réseau aient connaissance de cette répartition. En effet, les nœuds terminaux du réseau doivent attendre le beacon de *leur* coordinateur pour avoir connaissance de la répartition temporelle du médium pour la supertrame qui débute à la réception de ce beacon. Or, dans notre modification, nous avons supprimé la notion de CAP<sup>◇</sup> et de CFP, si bien qu'un slot non annoncé comme « réservé » pourrait être supposé comme librement utilisable dans le cadre d'un accès en mode *best-effort*. Il est bien évident que ce raccourci ne doit pas être fait sous peine d'entraîner l'utilisation en mode aléatoire d'un slot réservé dans une autre étoile, et ceci en toute bonne foi. Notre réseau se voit donc dans l'obligation d'appliquer par défaut une politique d'accès au médium de type déterministe, c'est-à-dire que *tout slot non renseigné est a priori réservé* et non dédié au mode *best effort*, comme c'est le cas avec 802.15.4.

Le coordinateur d'une étoile devra donc rappeler dans chacune de ces balises quels sont les slots réservés pour ses nœuds terminaux, quels sont les slots libres mais inutilisables car réservés dans une étoile voisine, et enfin quels sont les slots utilisables via CSMA/CA. Chaque slot devra entrer dans une de ces trois catégories, ou à défaut, être considéré comme non utilisable (deuxième catégorie).

### 2.3.9 Optimisation de l'accès au médium : attribution de GTS simultanés

Un des problèmes majeurs de notre méthode d'accès est l'optimisation du débit offert, comme dans toute méthode d'accès, en particulier celles basées sur TDMA<sup>◇</sup>. Afin de permettre un débit plus important, nous proposons de mettre en place un dispositif protocolaire permettant aux coordinateurs de pouvoir attribuer un GTS dans le même slot temporel qu'un autre GTS déjà attribué par un autre coordinateur, dans une autre étoile, seulement si ces deux étoiles sont « suffisamment éloignées l'une de l'autre ». Cette fonctionnalité va permettre la réutilisation des slots temporels pour permettre au réseau de prendre en charge plus de trafics. Nous appellerons par la suite ces GTS simultanés SGTS<sup>◇</sup>, pour « *Simultaneous GTS* ».

#### 2.3.9.1 Le principe

Pour pouvoir réaliser un accès au médium simultané sans créer de collision, il est nécessaire que le destinataire d'un message ne soit pas gêné par les autres émissions simultanées, c'est à dire que l'ensemble des signaux tiers ne perturbent pas le signal que l'on souhaite recevoir. En pratique, les modulations utilisées dans les transmissions numériques se prêtent assez bien à cette pratique, il suffit pour un récepteur que l'énergie reçue des perturbateurs soit inférieure à celle reçue de l'émetteur souhaité, en tenant compte d'une certaine marge.

Une série de mesures réalisées sur notre réseau prototype nous a permis de mettre en évidence les conditions de réception pour pouvoir pratiquer deux émissions simultanées sans entraîner de collision. Ces mesures seront présentées dans le chapitre 4 (cf. paragraphe 4.3.2.3 du chapitre 4, page 139).

#### 2.3.9.2 Nécessité d'une négociation protocolaire

Dans le cadre de notre méthode d'accès déterministe, l'affectation d'un SGTS doit faire l'objet d'une négociation afin de ne pas rompre avec le déterminisme de la méthode d'accès. En effet, si le concept

de GTS permet le déterminisme par une garantie d'absence de collisions, l'introduction des SGTS ne doit pas affaiblir les mécanismes présentés jusque là. Par ailleurs, la négociation est nécessaire car les propriétés relatives à la couche physique (puissances émises, sensibilités des récepteurs, bruit perçu, etc.) et aux antennes (rayonnement, omnidirectivité, gain, etc.) ne permettent pas de supposer la symétrie des transmissions : si un nœud *A* reçoit faiblement les transmissions d'un autre nœud *B*, il serait une erreur de supposer que *B* reçoit *A* avec la même faiblesse.

Illustrons l'idée de cette négociation par un exemple : considérons deux coordinateurs *A1*, *B1* et deux nœuds *A2*, *B2*. *A2* est rattaché à *A1* et *B2* est rattaché à *B1*. La topologie du réseau ainsi formé est représentée dans la figure 3.8. *A1* reçoit des messages émis par *A2* dans un GTS dédié à *A2*, *B1* reçoit des messages émis par *B2* dans un GTS dédié à *B2*.

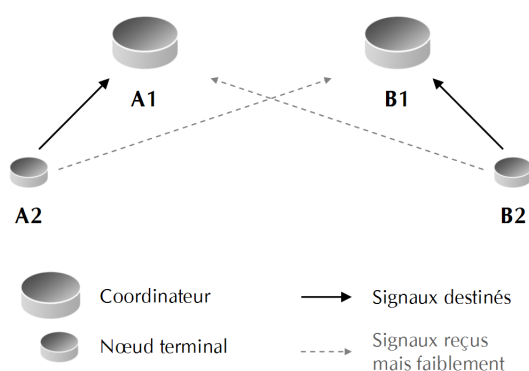


FIG. 3.8 – Topologie d'école du réseau illustrant la négociation d'un SGTS

Le principe de la négociation protocolaire est le suivant :

1. Le coordinateur *A1* constate que le signal émis par le nœud *B2* est reçu très faiblement, voire inaudible pour lui. Le GTS dédié à *B2* pourrait donc être réutilisé par *A2* et devenir un SGTS commun aux deux nœuds *A2* et *B2*. Le coordinateur *A1* va donc demander au destinataire des messages de *B2* – c'est à dire *B1* – comment celui-ci reçoit les messages en provenance de *A2* ; cette demande est effectuée via le message **SGTS.request(A2,B2)**.
2. Le coordinateur *B1* reçoit cette demande et écoute alors *A2* ; il compare la force des signaux reçus de *A2* et *B2* et renvoie une réponse à *A1* par le message **SGTS.response(A2,B2)**.
  - Si la réponse est favorable, les deux GTS peuvent être changés en SGTS (deux fois plus de débit offert), ou regroupés en un même SGTS (même débit offert mais le médium est moins sollicité).
  - Si la réponse est défavorable, *A1* peut soit décider d'en rester là, soit demander à *A2* de diminuer sa puissance d'émission et relancer le processus de négociation avec *B1*.

Ce principe est illustré par la figure 3.9. Notons que les conclusions de la négociation ne sont valables que sur un temps très court car les conditions de réception peuvent évoluer à tout instant, surtout si les nœuds concernés par le SGTS sont mobiles. Il faudra veiller à renégocier régulièrement le SGTS pour la continuité du service déterministe. En pratique, nous pourrions favoriser les négociations de SGTS en bloquant les émissions de nœuds terminaux à faible puissance, ou tout du moins à une puissance minimale pour assurer une bonne communication avec leur coordinateur. De même, les messages envoyés par les coordinateurs à leurs propres nœuds terminaux peuvent eux aussi se faire à faible puissance : les coordinateurs peuvent négocier des SGTS entre eux. Nous discuterons de cette éventualité dans le chapitre 4, à la lumière des résultats des mesures pratiquées sur notre réseau prototype.

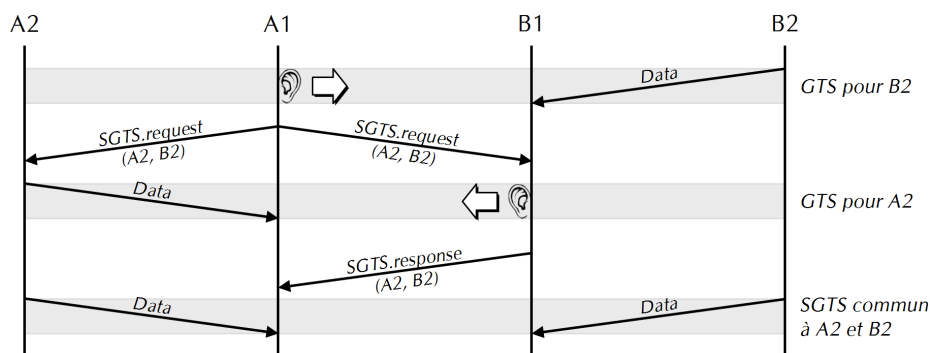


FIG. 3.9 – Principe de la négociation protocolaire d'un SGTS

### 2.3.9.3 Application à notre topologie

Dans le cadre de notre topologie, le principe de la négociation reste le même. Cependant, comme la gestion des ressources déterministes est confiée au supercoordonateur, la négociation des SGTS doit être validée par ce nœud. De plus, les contraintes liées à la topologie du réseau font qu'un nœud ne peut pas toujours s'adresser directement à un autre nœud.

La figure 3.11 illustre le déroulement temporel de la négociation d'un SGTS avec le relayage de la négociation par le supercoordonateur du réseau. Dans cet exemple, le Coordinateur 2 négocie un SGTS commun à (Noeud 1, Noeud 2) avec le Coordinateur 1, alors que les deux coordinateurs ne sont pas à portée radio l'un de l'autre. La réponse de la négociation est donnée par le supercoordonateur dans le message *SGTS.indication*. La figure 3.10 illustre la topologie de ce réseau.

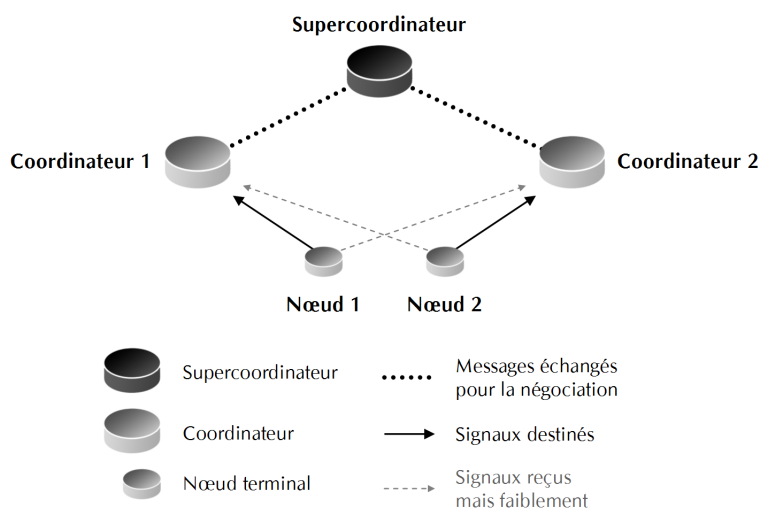


FIG. 3.10 – Topologie du réseau pour la négociation d'un SGTS



FIG. 3.11 – Principe de la négociation d'un SGTS dans le cadre de notre topologie

## 2.4 Implémentation du protocole : intégration dans une pile IEEE 802.15.4

La gestion de la couche MAC doit être modifiée pour pouvoir implémenter les nouvelles fonctionnalités proposées. La structure de certains messages doit être modifiée, de nouveaux messages sont créés et de nouvelles primitives vont être proposées à la liste existante.

Les modifications suivantes sont proposées sur les messages :

- D'une manière générale, l'introduction des fonctionnalités liées au déterminisme et aux niveaux de réservation  $n$  va introduire une modification de la structure des messages relatifs à l'association au réseau et à la gestion des GTS. De nouveaux champs devront être ajoutés dans les messages existants (`association.request`, `association.response`, `GTS.request`) pour permettre les associations déterministes et les demandes de GTS avec un niveau de réservation  $n \neq 0$ .
- Dans la mesure où les GTS (idem PDS, GBS, SGTS) peuvent avoir une fréquence inférieure à une supertrame (une sur deux, une sur quatre, etc.), il est nécessaire de prévoir un message de réponse à la requête `GTS.request`. Dans la norme actuelle, ce message n'est pas nécessaire car la réponse est donnée immédiatement par le coordinateur dans le beacon suivant (il contient/ne contient pas le GTS réclamé). Dans notre proposition, nous prévoyons l'introduction d'un message `GTS.response` afin de donner immédiatement une réponse au nœud demandeur, même si le GTS demandé avec un niveau 3 sera annoncé « 7 beacons plus tard ». De plus, dans notre topologie à plusieurs niveaux (nœuds terminaux – coordinateur, coordinateurs – supercoordinateur), il est nécessaire de prévoir un message de réponse pour les dialogues entre supercoordinateur et coordinateurs.
- Le protocole pour la négociation des SGTS nécessite lui aussi l'introduction de nouveaux messages pour le dialogue entre les coordinateurs concernés par la négociation. Deux nouveaux messages `SGTS.request` et `SGTS.response` doivent être introduits pour la réalisation du dialogue.

De même que pour les messages, certaines primitives vont être modifiées et d'autres vont être créées :

- Les primitives
  - `MLME-ASSOCIATE.request`,
  - `MLME-ASSOCIATE.indication`,
  - `MLME-ASSOCIATE.response`,
  - `MLME-ASSOCIATE.confirm`,
  - `MLME-GTS.request`,
  - `MLME-GTS.indication`,
  - `MLME-GTS.confirm`,
  - `MLME-COMM-STATUS.indication`

doivent être modifiées pour prendre en compte le côté totalement déterministe de la méthode d'accès et les niveaux de réservation  $n$  des GTS, PDS, GBS et SGTS.

- Une primitive `MLME-GTS.response` doit être introduite pour la génération du message `GTS.response`.
- Quatre primitives relatives à la gestion des SGTS sont proposées : `MLME-SGTS.request`, `MLME-SGTS.indication`, `MLME-SGTS.response` et `MLME-SGTS.confirm`. Ces quatre primitives seront appelées pour la génération des messages relatifs à la négociation des SGTS entre deux coordinateurs.

Toutes ces modifications nécessitent l'introduction de champs supplémentaires dans les messages de niveau MAC de 802.15.4. Nous ne détaillerons pas plus ces modifications : ce travail de développement sort du cadre de cette thèse et sera présenté en perspectives.

### 3 Conclusion

Le protocole qui vient d'être proposé répond aux critères énoncés en début de chapitre. Il propose une méthode d'accès totalement déterministe (totalement car même les phases d'entrée et de sortie du réseau peuvent se faire de manière déterministe) tout en assouplissant le mécanisme de gestion des GTS, déjà présents dans la norme IEEE 802.15.4.

Cependant, toutes les propositions faites dans ce chapitre doivent faire l'objet d'une validation. Le chapitre 4 présente cette ultime étape et les résultats qualitatifs et quantitatifs qui en ont été tirés. Plusieurs études ont été réalisées à cette occasion utilisant les Réseaux de Pétri, la simulation et le prototypage.



# Bibliographie du chapitre 3

- [IEE2 03] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY– *802.15.4 IEEE Standard for Information technology, Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)* – IEEEStd 802.15.4-2003 (2003)
  - [IEE2 03] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY– *802.15.4 IEEE Standard for Information technology, Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)* – IEEEStd 802.15.4-2003 (2003)
  - [LLI2 06] J.F. LLIBRE, P. PINEL et E. CAMPO – *Quel choix de source d'énergie pour rendre un système communicant autonome ?* – XII<sup>ieme</sup> Colloque National de la Recherche dans les IUT, Brest, France (2006)
  - [THOM 05] J.P. THOMESSE – *Fieldbus Technology and Industrial Automation – 10th IEEE Conference on Emerging Technologies and Factory Automation (ETFA 2005)*, Catania, Italie (Septembre 2005)
  - [VAL 05] T. VAL et G. JUANOLE – *La Qualité de Service dans les réseaux sans fil* – École d'été Temps Réel (ETR'05), Nancy, France (Septembre 2005)
  - [VDB 06] A. VAN DEN BOSSCHE, T. VAL et E. CAMPO – *Proposition of a full deterministic medium access method for wireless network in a robotic application – 63rd IEEE Vehicular Technology Conference (VTC-2006-Spring)*, Melbourne, Australie (Mai 2006)
  - [VDB 07] A. VAN DEN BOSSCHE, T. VAL et E. CAMPO – *Une méthode d'accès totalement déterministe pour un réseau personnel sans fil* – 8<sup>ieme</sup> congrès de l'Ecole Doctorale Systèmes de Toulouse (EDSYS'07), Albi, France (Mai 2007)
  - [ZBAL 05] ZIGBEE ALLIANCE – *ZigBee Specification* – ZigBee Document 053474r06, version 1.0 (2005)
-



## Chapitre 4

# Modélisation, validation et prototypage du protocole

*Après la présentation d'un état de l'art (chapitre 1), du réseau IEEE 802.15.4/ZigBee et de l'identification de ses faiblesses (chapitre 2), nous avons proposé, dans le chapitre 3, une série d'améliorations pour ce standard transposables à tout LP-WPAN. Dans ce chapitre 4, nous proposons de valider nos propositions par différentes méthodes complémentaires.*

---

<b>1</b>	<b>Introduction : une démarche de validation multi-outils . . . . .</b>	<b>111</b>
<b>2</b>	<b>Modélisation et validation formelle du protocole . . . . .</b>	<b>112</b>
2.1	L'outil Réseaux de Petri . . . . .	112
2.2	Présentation du modèle proposé . . . . .	112
2.2.1	Algorithme général déduit du protocole . . . . .	112
2.2.2	Un modèle de type <i>Père/Fils</i> . . . . .	114
2.2.3	Le modèle Réseaux de Petri . . . . .	115
2.2.3.1	Étude du processus père . . . . .	116
2.2.3.2	Étude du processus fils . . . . .	116
2.3	Validation formelle du modèle et analyse des résultats . . . . .	116
2.3.1	Présentation de l'outil TINA . . . . .	116
2.3.2	Analyse des résultats . . . . .	117
2.4	Conclusion de l'étude de validation formelle . . . . .	117
<b>3</b>	<b>Simulation : outils et résultats . . . . .</b>	<b>117</b>
3.1	Développement d'un outil de simulation . . . . .	117
3.1.1	Présentation de l'outil . . . . .	118
3.1.1.1	Mécanisme implémenté et simulé . . . . .	118
3.1.1.2	Fonctionnement du simulateur développé . . . . .	119
3.1.1.3	Utilisation du simulateur : descriptif de la topologie et exploitation des données . . . . .	119
3.1.1.4	Paramètres fixés pour la simulation . . . . .	120
3.1.2	Analyse des résultats obtenus . . . . .	121
3.1.2.1	Étude de l'association déterministe d'un coordinateur . . . . .	121
3.1.2.2	Étude de l'association déterministe d'un nœud . . . . .	123
3.2	Les travaux réalisés sous NS2 . . . . .	125
3.2.1	Présentation de l'outil . . . . .	125
3.2.2	Les modèles 802.15.4 / ZigBee existants sous NS2 . . . . .	125
3.2.3	Un résultat obtenu avec NS2 : débit utile dans la CAP avec CSMA/CA . . . . .	126
3.3	Conclusion sur les travaux de simulation . . . . .	127
<b>4</b>	<b>Prototype et métrologie . . . . .</b>	<b>127</b>

4.1	Présentation de l'existant . . . . .	127
4.1.1	Les plateformes IEEE 802.15.4/ZigBee existantes . . . . .	127
4.1.2	Modification du firmware existant . . . . .	129
4.2	Réalisation d'outils de tests et de mesures . . . . .	130
4.3	Conception et réalisation d'une pile prototype . . . . .	131
4.3.1	Évaluation des performances de la pile prototypée . . . . .	131
4.3.1.1	Évaluation de la sensibilité du récepteur . . . . .	131
4.3.1.2	Évaluation du temps de traitement d'un paquet . . . . .	132
4.3.1.3	Débit utile maximum . . . . .	133
4.3.1.4	Évaluation de la qualité de la synchronisation entre un co- ordinateur et ses nœuds . . . . .	134
4.3.1.5	Conclusion . . . . .	135
4.3.2	Validation de notre proposition par prototypage . . . . .	136
4.3.2.1	Évaluation de la qualité de la synchronisation entre le su- percoordonateur et ses nœuds . . . . .	136
4.3.2.2	Capacité maximale d'un slot . . . . .	137
4.3.2.3	Validation du concept de SGTS . . . . .	139
4.3.3	Conclusion sur les mesures réalisées sur le prototype . . . . .	142
4.4	Applications de démonstration . . . . .	142
4.4.1	Capteur de ligne ZigBee . . . . .	143
4.4.2	Passerelle IP/ZigBee . . . . .	144
4.4.3	Flotte de robots ZigBee . . . . .	144
4.4.4	Bilan sur les applications de démonstration . . . . .	145

---



# 1 Introduction : une démarche de validation multi-outils

(comme dans tout réseau industriel de capteurs)

Les règles protocolaires décrites dans le chapitre précédent nécessitent d'être validées, à la fois qualitativement et quantitativement. Dans ce double objectif, nous avons réalisé plusieurs travaux de vérifications et d'analyse des performances. Tout d'abord, le séquençement du protocole a été modélisé par Réseaux de Petri puis validé formellement grâce à des outils associés, afin de prouver qu'il ne souffre d'aucun blocage et que son exécution se cantonne bien à ce qui a été prévu. Cette première étude est présentée dans la section 2 « Modélisation et validation formelle du protocole » de ce dernier chapitre. Ensuite, le mécanisme d'association déterministe utilisant les PDS<sup>◇</sup> a été simulé dans le but de confirmer les performances temporelles qui avaient été déterminées théoriquement et présentées dans la section 2.3.4 du chapitre 3. Un simulateur logiciel spécifique a été développé pour l'occasion. Cette étude, ainsi que des travaux effectués sur le logiciel NS2, sont présentés dans la section 3 « Simulation : outils et résultats » de ce chapitre 4. Enfin, la méthode d'accès a été implémentée sur des véritables nœuds et une analyse de performances réelles sur prototype a été réalisée. Le concept novateur mais sensible de SGTS évoqué dans la partie 2.3.9 du chapitre 3 a lui aussi été validé par une série de mesures physiques et radio réalisées sur ce réseau prototype. Cette étude fait l'objet de la section 4 « Prototype et métrologie » de ce chapitre 4.

Le tableau 4.1 résume les différents points qui ont été validés par l'une de ces trois méthodes complémentaires et qui seront abordés dans ce chapitre.

	Méthodes		
	Validation formelle	Simulation	Prototypage
Séquencement de l'algorithme (exemple de l'association déterministe)	●		
Association déterministe par PDS et latence constatée pour les coordinateurs		●	
Association déterministe par PDS et latence constatée pour les nœuds		●	
Performances de l'existant avec l'accès en CSMA/CA dans la CAP		●	
Évaluation de la sensibilité du récepteur			●
Évaluation du temps de traitement d'un paquet par la pile SMAC			●
Synchronisation des nœuds d'une même étoile – gigue quantifiée			●
Synchronisation des nœuds du réseau par cascade de beacons – gigue quantifiée			●
Capacité maximale d'un slot et évaluation du débit garanti par la méthode d'accès			●
Caractérisation de l'effet de capture et validation des SGTS			●

TAB. 4.1 – Éléments validés par chaque méthode utilisée

Il est intéressant de noter que l'ensemble des contributions proposées d'un point de vue théorique dans cette thèse a été traité, que ce soit par validation formelle, par simulation ou par prototypage. Chaque étude est complémentaire car ces trois méthodes ne peuvent traiter l'ensemble des propositions faites dans notre contribution. Il aurait été intéressant de traiter, quand cela est possible, chaque point par deux ou trois techniques différentes. Ceci n'a pas été réalisé faute de temps, mais sera évoqué comme perspective intéressante en conclusion de la thèse.

Pour finir, la section 4.4 présente les différents projets qui ont été menés pendant la thèse et rendus

possibles par le travail réalisé sur ZigBee, sur IEEE 802.15.4, sur la nouvelle couche MAC et plus particulièrement sur le prototypage réel.

## 2 Modélisation et validation formelle du protocole

Cette première étape de validation prévoit de valider formellement le séquençement du protocole, c'est à dire de montrer par des méthodes analytiques que l'algorithme proposé se déroule toujours comme que nous l'avons prévu. Cette validation formelle est impossible à réaliser par des méthodes de simulation car elle nécessiterait de rejouer une infinité de scénarios ; seules les méthodes formelles peuvent démontrer ce type de propriété.

Le but de cette étape est donc d'assurer que, pour chaque exécution du protocole, il n'existe aucun cas de blocage (arrêt pur et simple du déroulement de l'algorithme et bouclage infini) ou de divergence (nécessité d'une mémoire infinie). Cette première étape est incontournable car il est nécessaire de valider ces propriétés *qualitatives* avant de pouvoir analyser *quantitativement* le prototype, que ce soit par simulation et/ou par prototypage. En effet, si le modèle souffrait d'un quelconque blocage, l'étude par simulation toute entière serait discréditée.

### 2.1 L'outil Réseaux de Petri

La technique des Réseaux de Petri [BRAM 83] [MERL 76] a été mise au point dans les années 60 par Carl Maria Petri. Les RdP constituent une technique de modélisation de systèmes à variables d'entrées, de sorties et d'états *discrets* dont le modèle peut-être représenté sous forme d'un graphe. Les Réseaux de Petri, et plus spécialement les Grafsets [MERC 06] qui en sont un cas particulier, sont populaires du fait de l'accessibilité et de la lisibilité de leur forme graphique qui rend compréhensible un système modélisé même pour une personne peu qualifiée. Cependant, les Réseaux de Petri n'en restent pas moins un outil mathématique très puissant.

Plusieurs variantes basées sur les fondamentaux de Carl Maria Petri ont été proposées par la suite : les *Réseaux de Petri Temporisés* [ATAM 94], les *Réseaux de Petri Stochastiques* [FLOR 85], les *Réseaux de Petri Temporisés Stochastiques* [MARS 85]... Ces variantes introduisent des grandeurs et des mécanismes additionnels (notions temporelles, probabilistes, etc.) qui permettent la modélisation de phénomènes impossibles à modéliser par les Réseaux de Petri originaux. Concernant nos travaux, le modèle proposé utilise les Réseaux de Petri simples.

### 2.2 Présentation du modèle proposé

#### 2.2.1 Algorithme général déduit du protocole

Le séquençement du protocole proposé dans le chapitre 3, c'est à dire l'échange des messages entre les différentes entités du réseau, peut se modéliser de différentes manières, selon le formalisme choisi et les objectifs attendus. Quelque soit le formalisme choisi, le principe général de la méthode d'accès proposée est le suivant :

1. attente de son slot déterministe (GTS, SGTS ou PDS si non encore associé),
2. émission de la demande dans le slot déterministe,
3. attente de la réponse, qui peut être largement différée si le niveau de réservation  $n$  est élevé.

Dans un premier temps, ce séquençement général a simplement été représenté sous forme algorithmique. A titre d'exemple, le séquençement d'*une association déterministe au réseau* (par utilisation des

PDS) est représenté sur la figure 4.1.

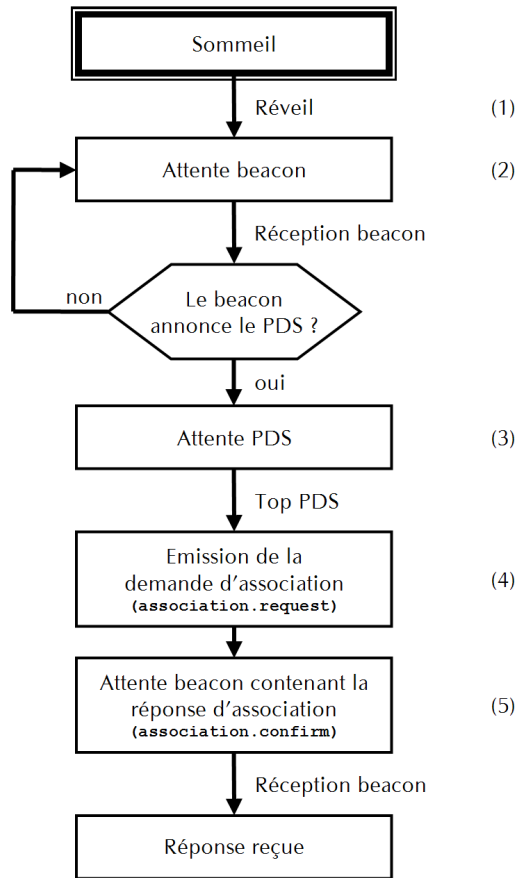


FIG. 4.1 – Séquencement du protocole pour une demande d'association d'un nœud au réseau

L'algorithme représenté illustre le comportement de tout nœud du réseau – coordinateur ou nœud terminal<sup>1</sup>. Il reprend exactement ce qui avait été décrit dans le chapitre 3 :

1. le nœud se réveille, il attend de recevoir le *beacon* de son entité supérieure,
2. le *beacon* reçu permet au nœud de se synchroniser sur son entité supérieure et d'entrer en mode *suivi de beacon*, c'est à dire la mise en veille rapide et le réveil automatique avant chaque *beacon*. Le nœud attend alors le *beacon* annonçant son PDS. Suivant le niveau de réservation  $n_{PDS}$ , ce *beacon* est plus ou moins fréquent,
3. le *beacon* reçu doit contenir des informations sur le PDS, le nœud patiente alors jusqu'à cet instant,
4. le nœud utilise son PDS pour émettre sa demande d'association (**association.request**) à son entité supérieure. Si la trame est reçue sans erreur, l'entité supérieure renvoie un acquittement de niveau 2, signifiant que la demande a bien été reçue (l'acquittement ne constitue pas une réponse

<sup>1</sup>Le supercoordinateur n'a pas de phase d'association puisqu'il est le premier à émettre. En revanche, dans le cadre d'une création complète du réseau, le supercoordinateur n'est pas nécessairement le premier nœud à se réveiller. Dans ce cas, *le réseau ne peut pas se former de manière déterministe, c'est à dire borné dans le temps*. L'ensemble du réseau est obligé d'attendre la réception des premiers *superbeacons* pour démarrer la phase d'association déterministe. Éventuellement, si le premier coordinateur éveillé ne reçoit aucun *superbeacon* dans une certaine fenêtre temporelle, il peut décider de devenir lui-même supercoordinateur ; cette fonctionnalité orientée *réseaux spontanés* pourra être discutée dans les perspectives de la thèse.

favorable).

5. le nœud attend la réponse (`association.confirm`) qui est annoncée dans le *beacon* suivant. Si la réponse est reçue et favorable, le nœud est accepté dans le réseau.

Notons que l'algorithme qui vient d'être commenté concerne bien la procédure d'association au réseau, mais la succession des étapes (synchronisation, attente du PDS, émission de la requête puis attente de la confirmation dans le *beacon* suivant) reste la même pour envoyer des données, demander plus de ressources déterministes, notifier un départ du réseau, etc. Les messages échangés sont différents mais la structure du dialogue reste la même.

### 2.2.2 Un modèle de type *Père/Fils*

La méthode d'accès que nous proposons s'articule autour d'une topologie hiérarchique à trois niveaux d'entités<sup>2</sup>, comme nous l'indiquions dans la section 2.3.1 du chapitre 3. La figure 4.2 illustre une topologie typique de notre réseau.

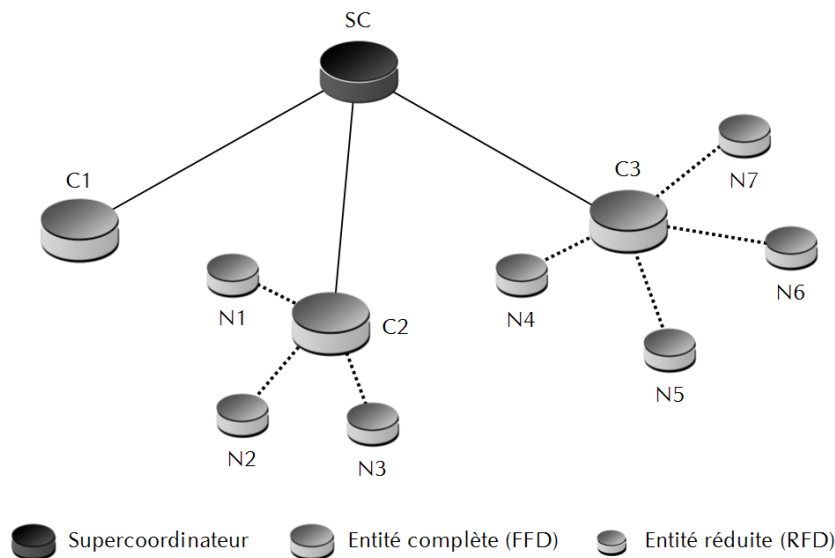


FIG. 4.2 – Éléments du réseau et liens entre ces éléments

Le réseau est constitué :

- d'un unique supercoordonateur (SC),
- d'un ou plusieurs coordinateurs (C1, C2 et C3), chacun maître de son étoile,
- de zéro, un ou plusieurs nœuds par étoile (N1, N2, N3, N4, N5, N6 et N7).

La description de la structure du réseau (éléments et liens entre ces éléments) nous permet d'établir une relation de type *père/fils* entre le supercoordonateur et le(s) coordinateur(s) d'une part, mais aussi entre un coordinateur et son (ses) nœud(s) rattaché(s). En effet, dans un cas comme dans l'autre, le père diffuse ses *beacons* sur lesquels le fils se synchronise et attend son PDS/GBS/GTS qu'il peut utiliser librement.

- si le fils est un coordinateur, il pourra utiliser son GBS pour diffuser lui-même un *beacon* et remonter une requête au supercoordonateur,
- si le fils est un nœud, il pourra utiliser son GTS pour demander plus de ressource ou envoyer une donnée.

<sup>2</sup>Nous pourrions généraliser cette structure à  $n$  niveaux, mais ce n'est pas l'objet de cette partie.

D'un point de vue structurel, la relation entre le supercoordonateur et les coordinateurs est la même que celle entre un coordinateur et ses nœuds. La seule différence se situe sur le plan temporel : les requêtes en lien avec le déterministe de l'accès au médium (association, demande de GTS, négociation de SGTS) doivent être acheminées jusqu'au supercoordonateur ; de ce fait, un délai supplémentaire est introduit dans le cas d'une communication nœud/supercoordonateur car l'information doit être relayée par le coordinateur.

Dans le cas réel, la relation entre les trois types de nœuds impose, pour le coordinateur, un fonctionnement séquentiel *père puis fils*. En effet, une fois associé au réseau (donc au supercoordonateur), le coordinateur diffuse ses *beacons* et se comporte avec ses nœuds de la même manière que le supercoordonateur se comporte avec les coordinateurs.

### 2.2.3 Le modèle Réseaux de Petri

La figure 4.3 représente le Réseau de Petri élaboré pour modéliser le comportement du protocole dans la phase d'association déterministe au réseau. On retrouve les deux processus : le père, à gauche et le fils, à droite. Les transitions disposées entre les deux servent à les synchroniser. Ces transitions représentent l'éventuelle activité sur le médium radio.

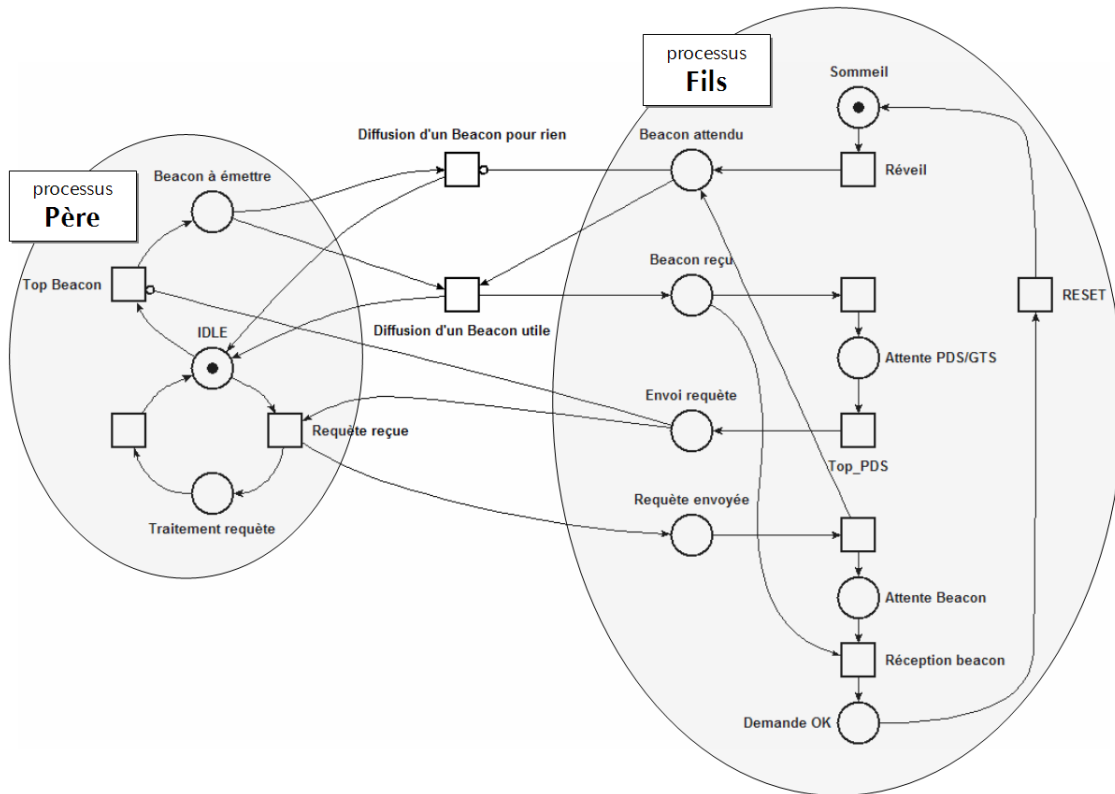


FIG. 4.3 – Représentation du modèle Réseau de Petri

### 2.2.3.1 Étude du processus père

Le processus père peut être dans 3 états :

- *inactif*,
- *en émission de beacon*,
- *en traitement de requête*.

Dans tous les cas, l'émission des trames *beacons* est régulière ; dans ce modèle non temporisé, les *beacons* sont émis tant que le fils n'a pas de requête à envoyer (présence de l'arc inhibiteur sur la transition *Top\_Beacon*). Dans la réalité, l'émission des *beacons* est déclenchée par un *timer* dont la période dépend du paramètre *BO* de IEEE 802.15.4 (cf. paragraphe 2.3.1.2 du chapitre 2).

Lorsque qu'un *beacon* est émis, deux cas peuvent se présenter :

- soit le fils attend ce *beacon* (un jeton est présent dans la place *Beacon\_attendu*) ; à la réception de cette balise, le fils peut poursuivre son séquençement. On parle alors d'une diffusion de beacon *utile*,
- soit le fils n'attend pas de *beacon* (absence de jeton dans la place *Beacon\_attendu*) ; l'arc inhibiteur présent sur la transition *Diffusion\_d'un\_beacon\_pour\_rien* permet au père de poursuivre même si le fils n'a pas besoin de ce beacon. On parle alors d'une diffusion de beacon *inutile*.

### 2.2.3.2 Étude du processus fils

Le comportement du processus fils suit exactement l'algorithme qui a été présenté plus haut sur l'organigramme de la figure 4.1 (réveil et attente du *beacon* annonçant le PDS, attente du PDS, envoi de la demande d'association, attente de la réponse). Seule une transition *RESET* a été ajoutée pour permettre la réinitialisation du processus fils dès que l'association est finalisée. Sans cette transition, le Réseau de Petri n'est pas *vivant*, c'est-à-dire qu'il présente un caractère bloquant ne permettant pas une validation formelle.

## 2.3 Validation formelle du modèle et analyse des résultats

### 2.3.1 Présentation de l'outil TINA

Le modèle Réseau de Petri proposé a été validé par le logiciel TINA [BERT 04] [TINA]. TINA (*TIme petri Net Analyzer*) est une plateforme logicielle pour l'édition et l'analyse des Réseaux de Petri temporisés (*Time Petri Nets*). Elle est développée et maintenue par le groupe OLC (Outils Logiciels pour la Communication) [OLC] du LAAS/CNRS (Laboratoire d'Analyse et d'Architecture des Systèmes) de Toulouse [LAAS]. Ce logiciel est également utilisé au laboratoire ICARE-LATTIS par un doctorant en relation avec le LAAS [KHOU 06].

TINA est un outil puissant qui permet la vérification de nombreux aspects des Réseaux de Petri (aspects *borné*, *blocage*, *réinitialisation*). Il s'appuie sur les propriétés intrinsèques des RdP et propose notamment une fonctionnalité de *validation formelle* qui apporte la preuve mathématique que la propriété étudiée est vérifiée avec un niveau de confiance de 100%. Les fonctionnalités de TINA permettent de réaliser une étude temporelle du modèle par l'utilisation des *Réseaux de Petri Temporisés*, fonctionnalité que nous n'utiliserons cependant pas ici.

Avant d'utiliser TINA, d'autres outils logiciels en rapport avec les RdP ont été mis en œuvre, notamment HPSIM [HPSI] de H. ANSCHUETZ. Cependant, TINA a été retenu pour ses capacités d'analyses formelles des RdP.

### 2.3.2 Analyse des résultats

TINA nous a permis de valider les propriétés suivantes de notre modèle RdP :

- **Aspect Borné** : cette propriété concerne le nombre fini d'états de marquage des places du RdP. Cet aspect doit être le premier à être vérifié sans quoi la validation formelle des autres aspects n'a aucun sens. L'aspect non borné se caractérise par un nombre infini de jetons dans au moins une des places du RdP : il signifie que le modèle *diverge* ou que le système implémenté nécessitera une quantité de ressources anormalement élevée (mémoire, temps CPU, etc.).
- **Aspect Vivant** : cet aspect permet de déceler les portions de *code mort*, c'est à dire l'absence de vivacité de certaines places et/ou le blocage de certaines transitions du Réseau de Petri, pour tout marquage initial et accessible du réseau. L'absence de vivacité permet de mettre en évidence des portions de code qui ne sont jamais exécutées (donc de relever les erreurs de modélisation) et des situations où le système modélisé risque de *bloquer*.
- **Aspect Réinitialisable** : la réinitialisation du système permet de retrouver un état initial (marquage initial) en partant de n'importe quel état de son fonctionnement. Cet aspect est fondamental dans la validation des systèmes de type *automate* qui présentent un fonctionnement cyclique.

Ces trois aspects de notre modèle ont été validés avec succès par TINA.

## 2.4 Conclusion de l'étude de validation formelle

La validation formelle du séquençement constitue une première étape indispensable à la démarche de validation globale du protocole. Grâce à cette première étude, nous avons démontré par une méthode formelle que l'algorithme que nous proposons ne présente aucun défaut de séquençement. Dès lors que notre modèle semble correct sur le plan qualitatif, nous allons maintenant pouvoir simuler son déroulement et étudier les performances temporelles qui en découle. Ces résultats ont été obtenus par simulation et sont présentés dans la section suivante.

## 3 Simulation : outils et résultats

Cette partie présente les travaux que nous avons effectués par simulation. Un outil logiciel spécifique a été développé pendant la thèse ; son développement et les résultats obtenus sont développés dans une première partie. Des travaux menés sur le simulateur NS2 seront évoqués dans une seconde partie.

### 3.1 Développement d'un outil de simulation

Après la validation formelle du séquençement du protocole d'accès déterministe proposée par les Réseaux de Petri, nous avons conçu un simulateur logiciel implémentant l'algorithme de distribution des GBS/GTS, algorithme prenant en compte les périodes d'attente imposées par les cycles prévus par notre protocole. Grâce à cette fonctionnalité, le simulateur peut délivrer des résultats sur la latence engendrée par la méthode d'accès au médium et permet ainsi l'étude des performances temporelles de la méthode d'accès proposée.

### 3.1.1 Présentation de l'outil

Le logiciel de simulation proposé implémente d'une part l'algorithme de distribution des PDS, GBS et GTS (qui, en réalité, sera placé dans le supercoordonateur) et d'autre part le système d'annonce de cette distribution par cascade de *beacons*. Il a été programmé en langage C et fonctionne sur une plateforme GNU/Linux. Son but est d'évaluer les performances temporelles d'un réseau utilisant uniquement la méthode d'accès déterministe (c'est à dire uniquement les PDS/GBS/GTS – aucun échange de données en CSMA/CA). Les performances temporelles déduites de cette étude peuvent être considérées comme le minimum garanti, les nœuds pouvant obtenir des retours « plus rapides » s'ils pratiquent également des accès au médium avec contention, mais alors sans garantie.

Il est à noter que le simulateur présenté ici établit des communications qui se basent sur une couche physique idéale : notre simulateur n'introduit ni délai de propagation, ni perte de paquets, ni erreurs de transmission. Nous insistons ici sur le fait que le but de cette étude n'est pas la simulation totale de la solution proposée, mais simplement une simulation de l'algorithme de distribution et de répartition des temps de parole sur le médium et des temps de latence pour le transport des données qui en découle.

#### 3.1.1.1 Mécanisme implémenté et simulé

Le mécanisme implémenté dans le simulateur est volontairement très proche du modèle qui avait été validé par utilisation des Réseaux de Petri. Il inclut la structure à trois niveaux (supercoordonateur, coordonateur, nœud) et ajoute la notion temporelle qui va permettre l'obtention de paramètres de performances. L'algorithme implémenté est le suivant :

- A  $t = 0$ , le Supercoordonateur est mis en fonctionnement. Il connaît les besoins temporels des nœuds considérés comme critiques, c'est-à-dire qu'il sait qu'un nœud A doit pouvoir prendre la parole au moins toutes les 100 ms (nécessité imposée par l'application), un nœud B au moins toutes les 400 ms, etc. Il prépare alors une répartition des PDS pour les  $2^{n_{MAX}}$  supertrames à venir<sup>3</sup> de manière à ce que chaque nœud puisse accéder au médium selon ses besoins et demander son association au réseau de manière déterministe. Notons ici que tant qu'aucune demande de ressource déterministe n'est reçue, cette répartition est conservée, indéfiniment.

Une fois la répartition des PDS effectuée (qui est instantanée dans la simulation), le Supercoordonateur commence la diffusion régulière des *superbeacons*.

- Les nœuds, qu'ils soient coordonateurs ou nœuds terminaux, se réveillent à un instant  $t$  tiré aléatoirement. Dès leur réveil, les nœuds cherchent à se synchroniser sur les balises de leur père (*superbeacons* pour les coordonateurs, *beacons* de leur coordonateur pour les nœuds terminaux). Si le père n'est pas encore réveillé, le fils attend son réveil.
- A la réception du premier *beacon* émis par le père, le fils est synchronisé et entre dans le mode *suivi de beacon* permettant l'économie d'énergie (réveil juste avant la réception du *beacon*, écoute, traitement puis *somnolence* jusqu'au *beacon* suivant.).

Le fils écoute chaque *beacon* jusqu'à entendre celui qui annonce son PDS. La fréquence du PDS dépend du  $n_{PDS}$  associé à ce nœud. Tant que son PDS n'est pas annoncé, le fils reste en suivi de *beacon* et ne demande pas à s'associer<sup>4</sup>. A ce stade, il n'a toujours pas émis le moindre message sur le médium.

- A la réception de la balise émise par son père annonçant son PDS, le fils l'utilise pour annoncer

---

<sup>3</sup>Les  $2^{n_{MAX}}$  supertrames constituent la vision maximale du supercoordonateur.

<sup>4</sup>Cette restriction est liée à l'objectif de cette étude qui n'autorise que l'utilisation des slots déterministes. Dans la réalité, un nœud peut demander à s'associer immédiatement par le biais d'un accès au médium avec contention, mais, encore une fois, sans garantie.



son réveil. Il demande alors l'association au réseau par un message de type **Association\_Request**. Dans ce message, le fils peut demander à garder le même niveau de déterminisme ( $n_{GTS} = n_{PDS}$ ), ou bien demander un niveau différent (il demande alors un nouveau  $n_{GTS}$  qui pourra être refusé par le supercoordonateur si la capacité du réseau ne le permet pas).

- Si le fils est un coordinateur, la demande est reçue directement par le supercoordonateur. De plus, les accès déterministes seront utilisés pour la diffusion de ses *beacons* : on parlera alors plutôt de  $n_{GBS}$  et non de  $n_{GTS}$ .
- Si le fils est un nœud terminal, la demande est reçue par le coordinateur du nœud, qui relait alors cette requête au supercoordonateur (introduction d'un délai supplémentaire qui sera quantifié dans la partie 3.1.2.2).
- Le supercoordonateur reçoit la demande d'association au réseau, directement (demande d'un coordinateur) ou indirectement (demande d'un nœud relayée par son coordinateur). Il prend en compte cette demande, confronte la valeur  $n_{GTS}$  ou  $n_{GBS}$  demandée avec la capacité restante du réseau et modifie (ou non) en conséquence la répartition qu'il avait préparée au départ. Il communique sa réponse dans le *superbeacon* suivant.
- Le nœud demandeur reçoit la réponse (directement si c'est un coordinateur ou indirectement, par son coordinateur, si c'est un nœud terminal). Si cette réponse est positive :
  - si le nœud est un coordinateur et s'il a obtenu le GBS demandé, il commence alors à diffuser ses propres *beacons*. Si certains de ses propres nœuds étaient en attente de ces *beacons*, ils sont alors synchronisés et attendent à leur tour leur PDS,
  - si le nœud est un nœud terminal et s'il a obtenu le GTS demandé, il utilise chaque GTS pour envoyer une trame de données quelconque.

### 3.1.1.2 Fonctionnement du simulateur développé

Le simulateur développé en Langage C s'articule très simplement autour d'une boucle infinie et d'une structure **switch/case**. Chaque état évoqué ci-dessus est codé dans un **case** et le passage d'un état à l'autre se fait quand les conditions de passage sont remplies. Chaque nœud est représenté dans la mémoire par une variable contenant son état. Au début du programme, un tableau de nœuds est initialisé avec tous les états à 0, le simulateur fonctionne avec un seul processus qui simule le fonctionnement de tous les nœuds du réseau.

L'aspect temporel est géré par une variable globale à tous les nœuds. Pour chaque nouvel instant, le comportement de chaque nœud est traité et le passage d'un état à un autre est testé. Lorsque tous les nœuds ont été traités, la variable temporelle est incrémentée d'un pas.

### 3.1.1.3 Utilisation du simulateur : descriptif de la topologie et exploitation des données

Avant de pouvoir utiliser le simulateur, la topologie du réseau et le besoin de chaque nœud doivent être décrits dans un fichier qui sera ensuite lu par le simulateur. La structure de ce fichier est simple : chaque ligne décrit un nouveau nœud et contient tous les paramètres caractérisant un nœud :

1. adresse,
2. type (coordinateur/nœud terminal),
3. adresse du père,

4. le niveau de réservation du médium au préalable ( $n_{PDS}$ ),
5. le niveau de réservation une fois associé ( $n_{GBS}$  si le nœud est un coordinateur ou  $n_{GTS}$  si c'est un nœud terminal).

Au cours de la simulation, chaque évènement est reporté sur l'écran ou dans un fichier de trace. Si la répartition des PDS, GBS, GTS évolue au cours du temps, le simulateur affiche cette nouvelle répartition sous forme de matrice, comme sur la figure 4.4. Cette représentation matricielle permet de voir rapidement comment le supercoordinateur a réparti l'ensemble des slots.

Cette matrice se lit ainsi :

- chaque colonne représente l'un des 16 slots de la supertrame IEEE 802.15.4,
- chaque ligne représente l'une des  $2^{n_{MAX}}$  supertrames à venir. Dans notre cas, nous avons fixé  $n$  à 3, soit une vision de 8 supertrames.
- chaque élément de la matrice représente donc l'un des  $16 * 2^{n_{MAX}}$  slots à venir,
- si le slot n'est pas affecté pour un accès déterministe (CAP), un point est représenté,
- si le slot est affecté à un nœud, l'adresse du nœud est indiquée,
- si le slot affecté est un GBS, l'adresse du nœud est notée entre { accolades }.

```
*** Ressources matrix at t = 7372800 ***
0 [ {1}  31    .    .    {3}    .    .    .    .    .    .    .    {2}    .    .    .    ]
1 [ {1}    .    .    .    {3}    .    .    .    .    .    21    .    {2}    .    .    .    ]
2 [ {1}    .    .    .    {3}    .    .    .    .    .    .    .    {2}    .    .    .    ]
3 [ {1}    .    .    .    {3}    .    .    .    .    .    21    .    {2}    .    .    .    ]
4 [ {1}  31    .    .    {3}    .    .    .    .    .    .    .    {2}    .    .    .    ]
5 [ {1}    .    .    .    {3}    .    .    .    .    .    21    .    {2}    .    .    .    ]
6 [ {1}    .    .    .    {3}    .    .    .    .    .    .    .    {2}    .    .    .    ]
7 [ {1}    .    .    .    {3}    .    .    .    .    .    21    .    {2}    .    .    .    ]
```

FIG. 4.4 – Exemple d'une répartition de PDS/GBS/GTS sous forme matricielle

Dès lors, on peut décoder la matrice représentée par la figure 4.4 ainsi :

- tous les slots 0 sont occupés par le nœud 1 (supercoordinateur) ( $n_{GBS} = 0$ ),
- les slots 1 de la première et de la quatrième supertrame sont occupés par le nœud 31 ( $n_{GTS} = 2$ ),
- tous les slots 4 sont occupés par le nœud 3 qui est un coordinateur ( $n_{GBS} = 0$ ),
- les slots 10 des supertrames 1, 3, 5 et 7 sont occupées par le nœud 21 ( $n_{GTS} = 1$ ),
- et enfin tous les slots 12 sont occupés par le nœud 2 qui est un coordinateur ( $n_{GBS} = 0$ ).

Lorsque tous les nœuds qui avaient été déclarés sont dans l'état « associé », le programme de simulation se termine et retourne la durée nécessaire à l'association pour chaque nœud du réseau. Ces valeurs retournées vont permettre l'étude de la durée nécessaire à l'association déterministe d'un nœud du réseau, que nous allons présenter dans la section suivante.

### 3.1.1.4 Paramètres fixés pour la simulation

Dans le cadre des travaux de simulation, certaines grandeurs, pourtant ajustables dans le programme de simulation, ont été figées. Ces valeurs, ainsi que les justifications pour les avoir figées, sont indiquées ci-après :

- conformément à ce qui est préconisé par la norme IEEE 802.15.4, la durée élémentaire d'un slot a été fixée à  $2^{BO} * 960 \mu s$ ,
- de même, toujours d'après la norme IEEE 802.15.4, notre supertrame conserve une longueur de 16 slots,

- $n_{max}$  a été fixé à 3, soit une vision de 8 supertrames pour le supercoordonateur,
- $BO$  a été fixé à 3 d'après les premiers résultats obtenus par prototypage, comme cela sera justifié dans la section 4.3.2.2.

Bien entendu, ces valeurs (constantes C, variables globales) peuvent être modifiées facilement dans le simulateur, même si elles ne sont pas modifiables directement par le fichier de configuration.

### 3.1.2 Analyse des résultats obtenus

Cette section présente les résultats obtenus grâce à notre simulateur. Dans un premier temps, nous nous intéresserons à la phase d'association déterministe au réseau d'un coordinateur, et plus précisément à la durée nécessaire à l'aboutissement de cette étape de création du réseau. Dans un second temps, nous nous intéresserons à la phase d'association déterministe d'un nœud.

#### 3.1.2.1 Étude de l'association déterministe d'un coordinateur

Cette première série de mesures met en évidence les performances temporelles de notre réseau simulé dans la phase d'association au réseau d'un coordinateur. Les étapes à valider sont les suivantes : réveil et synchronisation sur les *superbeacons*, attente du PDS, envoi de la demande d'association et d'un GBS puis attente de la réponse du supercoordonateur. Une première série de simulation a été effectuée et les résultats sont représentés sur la figure 4.5.

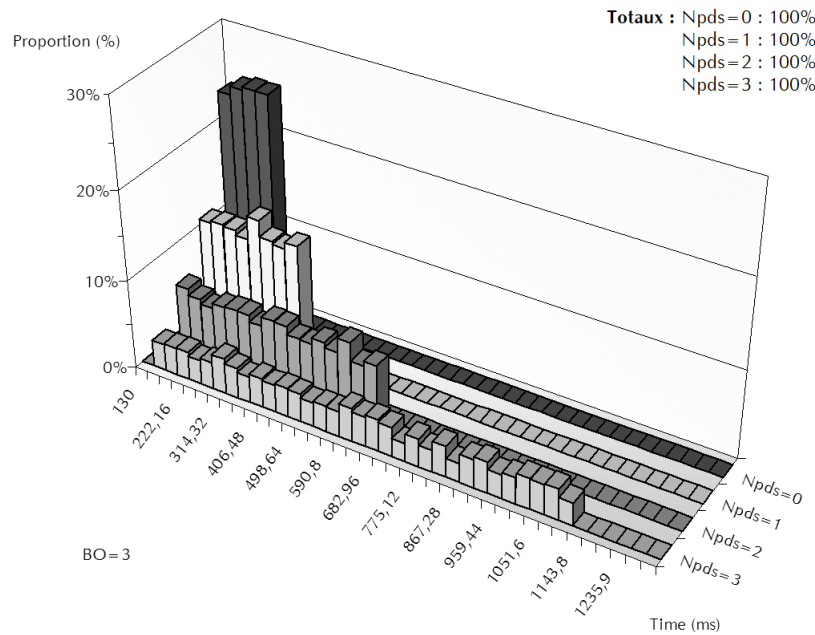


FIG. 4.5 – Représentation des fenêtres temporelles pour l'association déterministe d'un coordinateur (pour  $BO = 3$ )

La figure 4.5 représente les différentes fenêtres de temps pour l'association au réseau d'un coordinateur, pour plusieurs valeurs de  $n_{PDS}$  ( $0 \leq n_{PDS} \leq 3$ ). Notons que ce graphe a été obtenu grâce à la simulation de 2000 créations de réseau constitué d'un supercoordonateur et de 3 coordinateurs (soit 6000 mesures de temps d'association), pour un *Beacon Order* ( $BO$ ) de 3 (soit une émission de *superbeacon* toutes les 122,88 ms). Sur la figure 4.5, l'axe des abscisses représente le temps nécessaire à l'association

déterministe d'un coordinateur. Ce temps est découpé en intervalles car le système est temporellement discret – compte tenu du découpage en slots temporels, une demande d'association ne peut se faire qu'à certains instants. L'axe des ordonnées représente la proportion d'associations réalisées sur chaque tranche temporelle. Bien entendu, plus  $n_{PDS}$  est grand (c'est-à-dire plus la fréquence des slots réservés au préalable est faible), plus la plage temporelle sur laquelle les associations sont réparties est large.

A la vue de ces résultats, nous pouvons constater plusieurs points : tout d'abord, **nous sommes bien ici en présence d'une méthode d'association déterministe car 100% des demandes d'association au réseau obtiennent une réponse – qu'elle soit positive ou négative – en un temps qui a une borne maximale** : il est possible pour un coordinateur, à partir du moment où il entre sur le réseau, d'être certain qu'il obtiendra une réponse sur son association en moins de  $T_{ACmax}$  millisecondes (*Temps d'Association d'un Coordinateur MAXimum*), valeur qui dépend des paramètres  $BO$  et  $n_{PDS}$ , selon la formule :

$$T_{ACmax} = T_{slot} * 16 * 2^{BO} * (2^{n_{PDS}} + 1)$$

où  $T_{slot}$  est la durée élémentaire d'un slot, soit 960  $\mu s$  pour un slot PHY2450 de 802.15.4 ; la périodicité de la diffusion des *superbeacons* est alors de

$$T_{slot} * 16 * 2^{BO} = 15,36 \text{ ms} * 2^{BO}$$

D'après cette formule, et par exemple, pour  $BO = 3$  (comme c'est le cas sur la figure 4.5), un niveau de réservation  $n_{PDS} = 0$  permettra à un coordinateur d'envoyer sa demande d'association en moins de 122,88 ms après réception du *superbeacon*, un niveau 1 en moins de 245,76 ms, un niveau 2 en moins de 491,52 ms et un niveau de 3 en moins de 983,04 ms. Ces valeurs limites avaient été évoquées théoriquement dans la section 2.3.4 du chapitre 3 (cf. page 94). La simulation confirme ce qui avait été déterminé mathématiquement.

A la vue de ces résultats, on peut aussi constater que globalement, la distribution des temps d'association est uniforme, comme on pouvait s'y attendre, dans la mesure où, d'une part le protocole ne privilégie pas l'allocation au préalable de certains slots, et d'autre part, dans notre simulation, l'instant de réveil des nœuds suit une distribution uniforme et asynchrone vis-à-vis de « l'horloge du réseau ». Il n'y a donc aucune raison que certains slots soient privilégiés.

Enfin, on note qu'aucune association de coordinateur n'aboutit en moins de 130,56 ms ; ceci était attendu car le processus d'association au réseau d'un coordinateur nécessite une durée d'au moins 17 slots pour aboutir, quelque soit la valeur du  $n_{PDS}$ . En effet, dans le meilleur des cas, un coordinateur se réveillera juste avant l'émission du *superbeacon* annonçant son PDS, l'utilisera, et se saura associé au réseau à la réception du *superbeacon* suivant. On notera que si cette durée minimum  $T_{ACmin}$  (*Temps d'Association d'un Coordinateur MINimum*) ne dépend pas de  $n_{PDS}$ , en revanche, elle est directement fonction de  $BO$  ; dans notre cas où  $BO = 3$ , une durée de 17 slots vaut 130,56 ms comme on le déduit de la formule :

$$T_{ACmin} = T_{slot} * 17 * 2^{BO}$$

Cette première étude concernant l'association déterministe d'un coordinateur au réseau nous permet donc de conclure sur la propriété suivante : **grâce à la méthode d'accès que nous proposons, l'association au réseau déterministe d'un coordinateur est possible et répartie uniformément, et ce, dans une fenêtre temporelle bornée  $T_{AC}$  définie comme suit :**

$$17 * T_{slot} * 2^{BO} \leq T_{AC} \leq 16 * T_{slot} * 2^{BO} * (2^{n_{PDS}} + 1)$$

ce qui est impossible à borner en CSMA/CA.

### 3.1.2.2 Étude de l'association déterministe d'un nœud

A présent, nous nous intéressons à l'association déterministe d'un nœud. La principale différence entre l'association déterministe d'un nœud et celle d'un coordinateur est le retard engendré par le fait que la demande d'association nécessite d'être relayée par le coordinateur du nœud, puisque le nœud ne peut envoyer directement sa requête au supercoordinateur. De plus, l'étape de synchronisation du coordinateur sur le supercoordinateur doit avoir abouti avec succès.

Comme dans le paragraphe précédent, nous étudions cette phase pour plusieurs niveaux de réservation, mais pour cette étude, il est nécessaire de différencier le niveau de réservation des PDS du nœud ( $n_{PDS}$ ) du niveau de réservation des GBS du coordinateur ( $n_{GBS}$ ). En effet, dans ce cas, les paramètres à prendre en compte sont plus nombreux.

Le réseau simulé est constitué ainsi :

- 1 supercoordinateur diffusant ses *superbeacons* dans chaque supertrame,
- 4 coordinateurs diffusant leurs *beacons* dans chaque supertrame ( $n_{GBS} = 0$ ),
- 4 nœuds, chacun rattachés à un coordinateur, qui se voient affectés un niveau de réservation préalable variable, compris entre 0 et 3 ( $n_{PDS} = 0, 1, 2$  ou 3).

Le simulateur ainsi paramétré donne les résultats représentés dans le graphique de la figure 4.6. Ce graphique représente les différentes fenêtres de temps pour l'association au réseau d'un nœud, pour  $n_{PDS} = 0, 1, 2$  ou 3. Ce résultat a été obtenu grâce à la simulation de 21000 associations de nœuds, pour un  $BO$  de 3 (soit une émission de *superbeacon* toutes les 122,88 ms). L'axe des ordonnées représente la proportion des associations finalisées en un temps indiqué en abscisse.

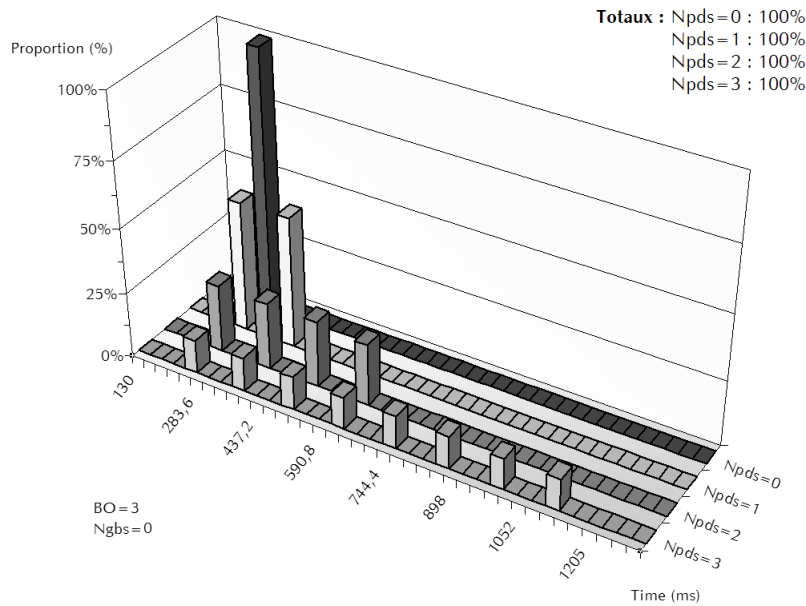


FIG. 4.6 – Représentation des fenêtres temporelles pour l'association déterministe d'un nœud terminal (pour  $BO = 3$  et  $n_{GBS} = 0$ )

L'interprétation des résultats de simulation nous permet de tirer plusieurs conclusions. Tout d'abord, là encore, nous sommes bien en présence d'une méthode d'association au réseau déterministe car 100% des nœuds obtiennent une réponse, qu'elle soit positive ou négative, dans une fenêtre de temps déterminée et bornée. La taille de cette fenêtre temporelle dépend des paramètres temporels du réseau : pour une certaine valeur  $n_{PDS}$  et une valeur  $n_{GBS}$  de 0 (le coordinateur du nœud diffuse un *beacon* dans chaque

supertrame), les associations sont réalisées en un maximum de  $2^{n_{PDS}} + 1$  supertrames. Temporellement, si nous considérons alors le paramètre  $BO$ , par exemple  $BO = 3$  comme c'est le cas pour le graphe de la figure 4.6 (une supertrame toutes les 122,88 ms), nous pouvons affirmer que 100% des associations seront réalisées en un temps maximal  $T_{ANmax}$  (*Temps d'Association d'un Noeud MAXimum*) qui suit la loi :

$$T_{ANmax} = T_{slot} * 16 * 2^{BO} (2^{n_{PDS}} + 1)$$

Cette équation peut être généralisée pour toute valeur de  $n_{GBS}$  selon :

$$T_{ANmax} = T_{slot} * 16 * 2^{BO} (2^{n_{PDS}} + 2^{n_{GBS}})$$

Cette généralisation est confirmée par le graphe de la figure 4.7 qui représente les résultats temporels de 1800 créations d'un réseau constitué d'un supercoordonateur et de 3 coordinateurs, chacun ayant un nœud. Pour cette simulation,  $n_{GBS}$  est fixé à 1 (soit une diffusion de *beacons* toutes les 2 supertrames) et  $n_{PDS}$  varie entre 1, 2 et 3.

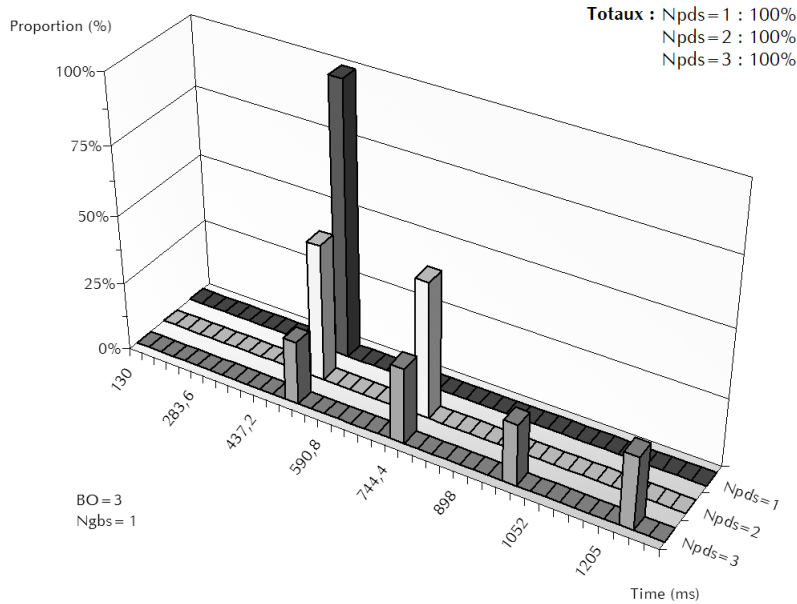


FIG. 4.7 – Représentation des fenêtres temporelles pour l'association déterministe d'un nœud terminal (pour  $BO = 3$  et  $n_{GBS} = 1$ )

Une seconde remarque concerne la durée minimale d'une association au réseau. En effet, sur le graphe de la figure 4.6, nous pouvons constater qu'aucune demande d'association n'est concrétisée en moins de 245,76 ms. Nous pouvons également constater que cette valeur ne dépend pas de la valeur de  $n_{PDS}$ . De même, sur la figure 4.7, aucune association n'est concrétisée en moins de 491,52 ms. Ceci s'explique par le fait que le mécanisme d'association d'un nœud nécessite au moins la transmission de trois *beacons* par son coordinateur, comme le dialogue protocolaire l'implique (rappel) :

1. le nœud attend de recevoir un premier *beacon* de son coordinateur qui annonce son PDS,
2. à la réception de ce *beacon*, le nœud attend son PDS,
3. au moment du PDS, le nœud émet sa demande d'association au réseau,
4. le coordinateur réceptionne cette demande, la répercute au supercoordonateur lors de l'émission d'un second *beacon*,

5. le supercoordonateur communique son approbation / désapprobation dans le *superbeacon* qui suit,
6. le coordonnateur relaie la réponse dans un troisième *beacon* et le nœud sait alors s'il est associé ou non.

De ce fait, le délai minimal pour l'association déterministe d'un nœud  $T_{NAmin}$  dépend uniquement des paramètres  $BO$  et  $n_{GBS}$  selon la loi :

$$T_{ANmin} = T_{slot} * 16 * 2^{BO+n_{GBS}+1}$$

Enfin, cette étude par la simulation nous montre que, comme dans le cas de l'association du coordonnateur, la répartition temporelle au sein de cette fenêtre est uniforme, car le protocole, dans cette version, ne privilégie pas certains créneaux temporels par rapport à d'autres.

Cette seconde étude sur l'association déterministe d'un nœud au réseau nous permet donc de conclure sur la loi suivante : **grâce à la méthode d'accès que nous proposons, l'association au réseau déterministe d'un nœud est possible et répartie uniformément, et ce, dans une fenêtre temporelle bornée  $T_{AN}$  selon la formule :**

$$T_{slot} * 16 * 2^{BO+n_{GBS}+1} \leq T_{AN} \leq T_{slot} * 16 * 2^{BO}(2^{n_{PDS}} + 2^{n_{GBS}})$$

ce qui, encore une fois, est impossible à borner avec une méthode d'accès basée sur CSMA/CA.

## 3.2 Les travaux réalisés sous NS2

### 3.2.1 Présentation de l'outil

NS (*Network Simulator*) [NS] est un simulateur à événements discrets destiné aux travaux de recherche sur les réseaux. Il permet la simulation d'une grande diversité de protocoles à tous les niveaux d'une pile protocolaire (protocoles de niveau *réseau* et *transport* comme TCP/IP, protocoles de routage et de multicast, ...). Il permet la simulation d'un grand nombre de réseaux locaux et/ou étendus, filaires ou sans fil.

Le projet NS a été lancé en 1989 comme une variante du simulateur *REAL network simulator*. Comme de nombreux projets ouverts, il évolue au gré des contributions de ses utilisateurs. Il a fini par s'imposer, depuis quelques années, comme une référence dans le domaine de la simulation des réseaux. La grande différence entre NS et ses principaux concurrents (OPNET [OPNE], Qualnet [QUAL 02] [QUAL]) est qu'il est totalement *Open Source* et gratuit. NS n'en reste pas moins un outil soutenu par des acteurs puissants de la recherche, comme le DARPA, Sun, Xerox, UCB, USC/ISI [VINT 95], le CNRS français...

Le projet NS propose un ensemble de logiciels pour la simulation des réseaux : le moteur de simulation NS2 en est l'élément central ; il génère des fichiers *trace* qui doivent être interprétés pour en extraire les résultats. D'autres outils comme NAM (*Network Animator*) permettent une exploitation plus visuelle et conviviale des résultats de simulation.

### 3.2.2 Les modèles 802.15.4 / ZigBee existants sous NS2

Lorsque nous avons débuté nos travaux de thèse, seul un modèle IEEE 802.15.4 / ZigBee [SAMS], [ZHEN 04], [ZHEN 06] existait pour NS2 et avait été publié. Ce modèle a été développé par J. ZHENG and Myung J. LEE du département *Electrical Engineering* de l'Université *City College* de New York en

collaboration avec SAMSUNG. Il a d'abord été publié sous forme de *patch* de NS2 avant d'être intégré dans la distribution officielle à partir de la version 2.28. Il proposait la plupart des fonctionnalités décrites dans le *draft D18* [IEE3 03] qui a précédé la norme actuelle. En revanche, parmi les mécanismes non implémentés se trouvaient notamment la gestion des GTS, ce qui, de notre point de vue, constituait un manque important. Un autre modèle réalisé par G. LU de l'équipe ANRG (*Autonomous Networks Research Group*) de l'Université *Southern California* de Los Angeles [GANG 04] avait aussi été développé mais de manière bien plus sommaire, car n'implémentant que la partie CAP de la MAC de 802.15.4.

Durant la thèse, nous avons tenté de compléter en vain le modèle de J. ZHENG. Par manque de documentation et d'expérience sur la programmation de ce modèle sous NS2, l'implémentation de la gestion des GTS – étape préliminaire nécessaire pour pouvoir comparer notre proposition à l'existant – n'a pu être finalisée.

Cependant, bien que les modèles existants soient incomplets sur la partie CFP/GTS de IEEE 802.15.4, il nous a semblé intéressant de reprendre ici quelques résultats obtenus par l'utilisation de NS2 sur la partie CAP et CSMA/CA. Ces résultats pourront être ensuite comparés avec les performances de notre méthode d'accès déterministe (cf. 4.3.2.2).

### 3.2.3 Un résultat obtenu avec NS2 : débit utile dans la CAP avec CSMA/CA

Cette étude, initiée par [GANG 04], a été réalisée par la simulation d'un réseau de 49 noeuds répartis uniformément sur un terrain de  $7 \times 7$ , tous espacés de 4 m, le coordinateur du réseau se trouvant au milieu. La portée étant fixée à 16 m, tous les noeuds sont à portée du coordinateur mais tous ne se trouvent pas dans le même domaine de diffusion<sup>5</sup>. Les paramètres temporels de la supertrame sont  $BO = 0$  et  $SO = 0$  (pas de somnolence cyclique), les paquets de données ont une taille de 50 octets et les *beacons*, 24 octets. L'étude porte sur deux cas : une unique source (pas de collisions possibles) ou 20 sources. Les résultats sont visibles sur la figure 4.8.

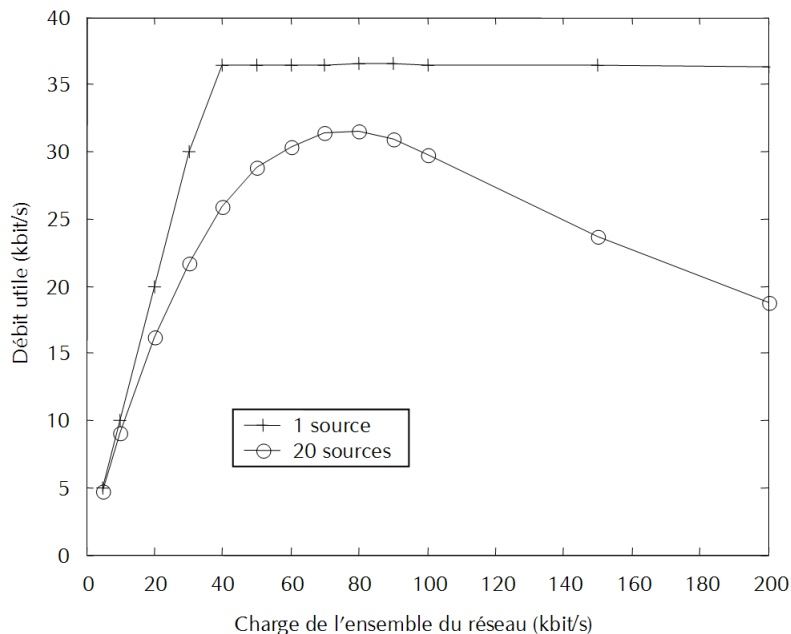


FIG. 4.8 – Débit utile avec CSMA/CA dans un réseau 802.15.4 avec *beacons*

Sur cette figure, nous pouvons constater que, pour une unique source, le débit utile maximum est de

<sup>5</sup>Le CSMA/CA implémenté dans 802.15.4 ne prévoit pas de RTS/CTS. Cette remarque est importante dans le cadre d'une étude de débit car lorsque deux noeuds trop lointains contactent en même temps le coordinateur, la collision est inévitable sans RTS/CTS.



38 kbits/s. Ce débit est très inférieur au débit *bande de base* de 250 kbits/s, ceci étant dû à la présence des temps inter-frames, des *backoffs* et des acquittements. Dans le cas de l'étude portant sur 20 sources, on retrouve bien la courbe « en cloche » typique de CSMA/CA, le maximum du débit utile étant de 32 kbits/s pour une sollicitation de 70 kbits/s globale au réseau. Au delà, les performances diminuent fortement, ce qui ne sera pas le cas dans le cadre de notre méthode d'accès déterministe, comme nous le verrons plus tard dans la section 4.3.2.2!

### 3.3 Conclusion sur les travaux de simulation

La conception du simulateur et l'analyse des résultats nous ont permis d'obtenir les premiers paramètres temporels de performances d'un réseau mettant en œuvre la méthode d'accès que nous proposons, spécifiquement dans les phases critiques que sont l'association au réseau et la demande de ressources déterministes. Bien entendu, le simulateur que nous avons développé n'est pas complet ; comme nous l'avons précisé au début de cette partie, en particulier, il ne considère pas les erreurs sur le médium.

Cependant, les résultats que nous obtenons grâce à notre simulateur sont prometteurs car le système simulé présente bien les caractéristiques, en terme de déterminisme, que nous attendions. La méthode d'accès, si nous la considérons indépendamment de tout le reste du système de communication, ne souffre d'aucun blocage et semble temporellement très efficace. Le déterminisme attendu est bien présent, car toute demande effectuée par un nœud du réseau, du point de vue de la répartition des accès concurrents sur le médium, obtient une réponse dans un temps imparti.

Suite à l'orientation du travail de la thèse et à l'absence de maturité sur les modèles existant (NS2), nous avons décidé de ne pas consacrer plus de temps à l'étude par simulation ; à défaut de le simuler, un travail plus fin mettant en œuvre un médium réel a été avantageusement réalisé par un prototype matériel et des métrologies avancées. Ce travail est exposé dans la section suivante.

## 4 Prototype et métrologie

La partie prototypage constitue le troisième volet de la validation de nos propositions. Cette partie vient compléter les deux précédentes en ajoutant les aspects imparfaits de la transmission, les caractéristiques intrinsèques du matériel et du logiciel et la portée radio limitée de nos *transceivers*. De plus, de part les activités applicatives et pédagogiques du laboratoire, l'aspect prototypage constituait, dès le départ, un objectif incontournable.

Nous nous sommes donc naturellement tournés vers la réalisation d'un prototype matériel, tout d'abord pour valider la faisabilité de notre proposition et en évaluer les performances réelles puis pour présenter un prototype fonctionnel de démonstration.

### 4.1 Présentation de l'existant

#### 4.1.1 Les plateformes IEEE 802.15.4/ZigBee existantes

Le début de la thèse (octobre 2004) a coïncidé avec la commercialisation des premiers kits de développement ZigBee et IEEE 802.15.4. Durant toute la période d'étude, il y a toujours eu deux principales (majeures) solutions proposées par les industriels :

- La solution CHIPCON, tout d'abord basée sur un microcontrôleur MICROCHIP, puis dernièrement associé à un microcontrôleur TEXAS INSTRUMENT,

- La solution FREESCALE, d'abord basée sur un microcontrôleur MOTOROLA 68HC089S, puis dernièrement intégrée sur un composant *monochip* du même fabricant.

Tous les travaux de prototypage élaborés pendant la thèse ont utilisé la seconde solution. Nous disposons d'un contact d'excellente qualité chez FREESCALE, à Toulouse, ce qui nous a largement permis de gagner en efficacité. Nous avons été amenés à travailler sur plusieurs plateformes de la famille FREESCALE :

- des cartes de développement (13192-SARD, première génération),
- des modules compacts (*ZigBee Reference Design*, ZRD01),
- des cartes de développement (13213-SRB et 13213-NCB, seconde génération).

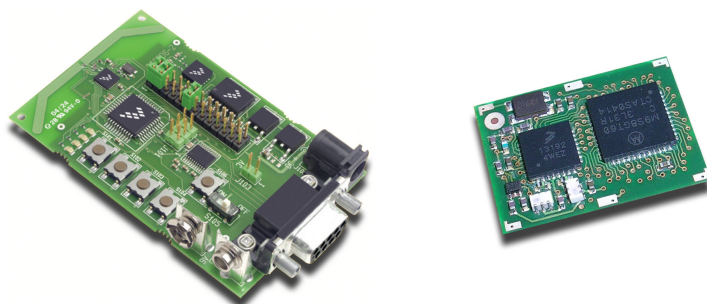


FIG. 4.9 – Quelques plateformes IEEE 802.15.4/ZigBee de FREESCALE

Ces trois plateformes matérielles sont équivalentes sur le plan de la mémoire (64 ko) et du CPU (8 bits, cadencé par le module radio). Elles disposent toutes d'un port série (RS-232 et/ou USB), d'un port de programmation BDM<sup>◇</sup> (port de programmation dédié) pour le téléchargement du *firmware* et de plusieurs GPIO<sup>◇</sup> « tout ou rien » et analogiques. Les composants de la carte (microcontrôleur, modem) fonctionnent sur une large plage d'alimentation (entre 2 et 3,6 volt) pour pouvoir utiliser aussi bien 2 piles alcalines (3,2 volt) que des accus NiCd ou NiMh en fin de vie (moins de 2,3 volt).

Les plateformes FREESCALE de première génération étaient équipées de deux *chips* : le microcontrôleur 8 bits (classique, de type MC9S08GT60) et le modem radio 2,4 GHz (MC13192, développé spécifiquement par FREESCALE pour équiper les nœuds ZigBee). Dans le cadre de cette solution, le microcontrôleur est cadencé par le modem qui peut ainsi gérer le mode *somnolence* ; le modem inclut un timer de très faible consommation, unique composant au travail dans ce mode. La dernière génération de plateformes FREESCALE est, quant à elle, monochip : le MC13213 renferme sous un même boîtier plastique un microcontrôleur 8 bits et un modem radio équivalent au MC13192, permettant ainsi au constructeur de proposer une solution plus compacte.

Le modem 2,4 GHz est chargé de l'émission et de la réception des *paquets* de niveau *physique*<sup>6</sup>, calcul du CRC inclus (aussi bien à l'émission qu'à la réception). Le modem communique avec le microcontrôleur via un bus SPI ; pour réaliser cette communication, deux modes sont envisageables :

- avec le mode *packet*, les données transitent sous forme de bloc et sont tamponnées dans le modem avant émission sur le médium ou avant envoi au microcontrôleur, ce qui implique, dans un sens comme dans l'autre, que toutes les données ont été reçues par le modem avant de pouvoir les traiter. Ce mode est le plus simple à mettre en œuvre mais induit une certaine latence proportionnelle à la longueur du paquet émis/reçu.
- avec le mode *stream*, les données transitent « au fil de l'eau ». En émission, le microcontrôleur signale au modem qu'une nouvelle trame de  $n$  octets va être envoyée et le modem commence à

<sup>6</sup>Nous avons vu dans le second chapitre que c'est par ce terme que 802.15.4 définit une quantité d'informations transmises au niveau physique, bien que cet emploi constitue un non sens du point de vue de l'OSI !

émettre le préambule, le fanion de départ et la longueur  $n$ . Le microcontrôleur envoie ensuite les données sur le bus SPI qui sont immédiatement envoyées sur le médium. En réception, l'opération est la même. Le mode *stream* est plus complexe à mettre en œuvre et induit une charge CPU plus importante ; en revanche, il induit une latence très faible et constante (elle ne dépend pas de la longueur de la trame reçue).

Le bus SPI permet également au microcontrôleur de configurer le modem MC13192 (changement de canal, choix de l'une des 18 puissances d'émission possibles, configuration du *timer* à faible consommation, etc.) ou de l'interroger (énergie sur le médium, puissance de la dernière trame reçue, etc.).

#### 4.1.2 Modification du firmware existant

Au niveau logiciel, les plateformes FREESCALE sont programmables via le port BDM ou via le port série si le module dispose d'un *bootloader*. Le *bootloader* est un micrologiciel exécuté à chaque démarrage du microcontrôleur qui permet notamment de flasher l'EEPROM sans port spécifique.. Toute la mémoire du microcontrôleur peut être reprogrammée : le programmeur dispose d'une grande liberté pour développer des applications qui peuvent interagir directement au niveau réseau, voire au niveau MAC, comme c'est le cas pour notre prototype<sup>7</sup>.

FREESCALE fournit trois solutions logicielles pour le programmeur (dans les dernières versions des kits de développement, FREESCALE parle de *Code Bases*) pour développer des applications communicantes :

- SMAC (*Simple Medium Access Control*) est la solution la plus simple. Elle propose un jeu de primitives très simplifié pour permettre le développement rapide d'applications communicantes propriétaires. Les applications basées sur SMAC ont une compatibilité 802.15.4/ZigBee au niveau de la couche physique, mais pas nécessairement aux niveaux supérieurs (formats des messages, mécanismes implémentés). Cette solution proposée par FREESCALE est gratuite et utilise très peu de ressources mémoire et CPU (l'ensemble du code de la pile SMAC tient sur 4 ko de mémoire).
- IEEE 802.15.4 propose l'ensemble des fonctionnalités prévues par le standard du même nom (couches *physique* et *liaison*). Nous verrons plus bas que, contrairement à ce qu'annonce le constructeur, toutes les primitives définies par l'IEEE n'ont pas été implémentées par FREESCALE – du moins, à l'heure où ces lignes ont été écrites – ce qui nous a posé quelques problèmes lors du développement du prototype. Comme SMAC, la pile 802.15.4 est elle aussi gratuite. La pile RFD tient sur 18 ko de mémoire et la pile FFD sur 40 ko.
- La solution ZigBee permet au programmeur de développer une solution réseau dont la pile a été certifiée par la *ZigBee alliance*. Toute la pile est fournie, complète, sous forme de librairies. Le développeur n'a plus qu'à développer le ou les profils correspondant à son application. A la différence de SMAC et 802.15.4, l'utilisation de la pile ZigBee est payante. La licence coûte environ \$1000 pour une version avec librairies et \$50.000 pour une version *open source*.

Au cours des travaux de prototypage réalisés pendant la thèse, nous avons rencontré des difficultés pour modifier la pile IEEE 802.15.4 fournie par FREESCALE. Grâce à C. ZARADER de FREESCALE Toulouse, nous avons pu obtenir le code de la pile 802.15.4 et nous avons entrepris une importante modification de la pile existante pour implémenter nos propositions et permettre la réalisation du prototype. Malheureusement, nous avons constaté que la partie sur les GTS n'était pas opérationnelle. De plus, le code réalisé par FREESCALE étant très compact et fortement optimisé, il était difficile de le modifier, même légèrement.

Nous avons finalement décidé de baser notre prototype sur la pile SMAC, en développant les fonctionnalités de 802.15.4 dont nous avons besoin pour le prototype (génération de *beacons* et synchronisation, envoi/réception de trames de gestion et de données, etc.). Dans nos perspectives, nous prévoyons le

---

<sup>7</sup>Cette particularité constitue une grande originalité dans le monde des réseaux, où, généralement, il est impossible de reprogrammer intégralement une couche MAC, du moins sur du matériel non spécifiquement désigné à cet usage.

développement d'un nouveau prototype plus complet.

## 4.2 Réalisation d'outils de tests et de mesures

Pour réaliser les mesures sur le prototype, nous avons exploité plusieurs fonctionnalités matérielles et logicielles de la plateforme et réalisé plusieurs outils.

- Pour les mesures à caractère temporel internes au module, nous avons utilisé l'un des *timer* du composant, permettant ainsi de dater des événements. Les résultats des mesures sont envoyés sur le port série puis exploités, en différé, sur un PC par un logiciel spécifique développé à cet effet.
- Pour les mesures temporelles mettant en jeu plusieurs modules, nous avons utilisé les GPIO parallèles du microcontrôleur : en assignant chaque événement à un bit de GPIO et par utilisation conjointe d'un analyseur d'état logique temporisé, nous disposons alors d'un outil de mesure temporel très performant et indépendant de notre méthode de synchronisation globale aux modules (ce qui nous a d'ailleurs permis de l'évaluer – cf. paragraphe 4.3.1.4).
- Pour les mesures sur le médium radio, nous avons réalisé deux outils dès le début de nos travaux :
  - un *module ZigBee espion*, c'est-à-dire un récepteur, bloqué sur un canal, qui envoie sur son port série le contenu de toute trame reçue sur la radio ; il est également doté d'un système de datation d'une précision de  $64 \mu s$  et d'un indicateur de puissance du signal reçu d'une précision de 0,5 dBm. Chaque trame reçue se voit associée son instant de réception et sa puissance de réception.
  - un *analyseur de trames* sur PC, facilement adaptable au rythme des modifications que nous apportons sur le protocole.

Bien entendu, ces deux outils fonctionnaient ensemble – lancement d'une capture avec l'espion et stockage des données arrivant sur le port série du PC, puis interprétation et visualisation des trames capturées grâce à l'analyseur de trames (comme illustré figure 4.10).

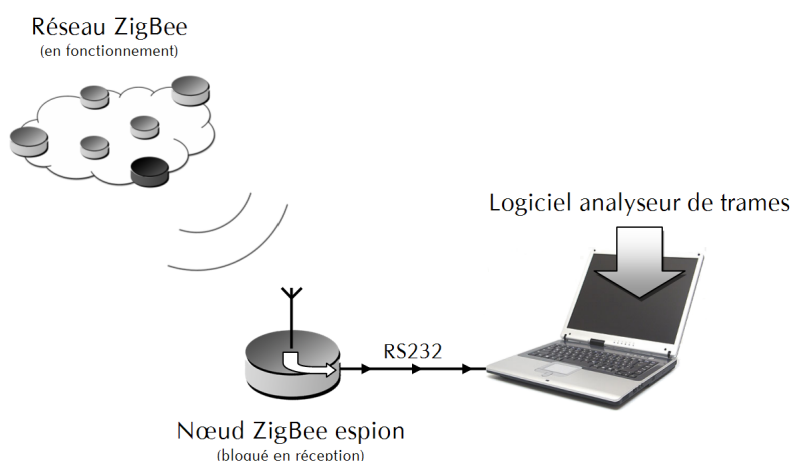


FIG. 4.10 – Utilisation conjointe du module espion et de l'analyseur de trames

Ces outils nous ont permis de faire les tests et les mesures nécessaires à la validation de notre pile prototype qui sont détaillés dans la section suivante.

## 4.3 Conception et réalisation d'une pile prototype

### 4.3.1 Évaluation des performances de la pile prototypée

Tout au long du développement du prototype, nous avons cherché à caractériser finement les performances de notre réalisation, depuis le matériel et le logiciel de base utilisé pour la réalisation, jusqu'à la validation de nos propositions.

Cette section présente les résultats des différentes mesures qui ont été réalisées pendant le prototypage de notre pile protocolaire : sensibilité du récepteur et taux d'erreur trame typique, temps de traitement typique par paquet, débit utile et qualité du mécanisme de synchronisation des noeuds par *beacons*.

#### 4.3.1.1 Évaluation de la sensibilité du récepteur

Le but de cette première série de mesures est d'évaluer la sensibilité du récepteur 2,4 GHz présent dans le MC13192 et d'évaluer le taux de perte de trame typique dans le cadre d'une utilisation idéale, c'est-à-dire sans perturbation électromagnétique et sans accès au médium concurrent (probabilité de collisions nulle).

Cette série de mesures a été réalisée dans la *chambre anéchoïque*<sup>8</sup> de l'IUT de Blagnac. Un programme de test se basant sur la pile SMAC a été développé et près de 30000 trames indexées ont été transmises pour réaliser cette série de mesures. La fonctionnalité d'ajustement de la puissance d'émission a été utilisée pour permettre d'évaluer la sensibilité du récepteur sur une large plage – depuis des conditions de réception très confortables ( $< -30$  dBm) aux conditions limites données par le constructeur ( $-95$  dBm). Les résultats sont présentés sur la figure 4.11.

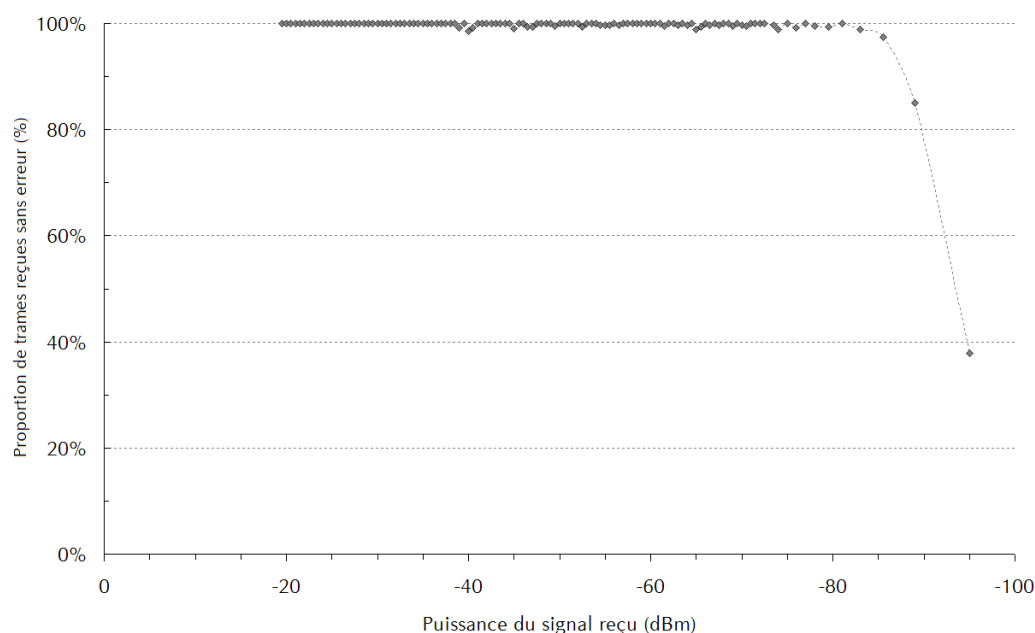


FIG. 4.11 – Évaluation de la proportion de trames perdues en fonction de la puissance du signal reçu

<sup>8</sup>Une chambre anéchoïque est une pièce étanche sur le plan électromagnétique et dont le revêtement intérieur supprime les réflexions sur les parois. C'est un lieu quasi idéal pour ce type de mesures car il permet de s'affranchir des nombreuses sources de rayonnement et de perturbations, notamment dans la bande des 2,4 GHz (points d'accès WiFi et Bluetooth, fours à micro-ondes, etc.).

Les résultats de cette étude permettent de montrer que, même dans un environnement idéal du point de vue des couches *physique* et *liaison* (pas de perturbations électromagnétiques, pas de méthode d'accès), la transmission n'est pas exempte d'erreurs, y compris dans des conditions de réception excellentes (quelques pertes constatées dans des conditions meilleures que -50 dBm). Il est intéressant de constater ici que, même si nos travaux proposent une méthode d'accès au médium déterministe, les erreurs introduites au niveau de la couche physique sont, elles, inévitables à notre niveau et doivent être prises en compte.

Nous pouvons également noter ici que les résultats de cette étude pourraient permettre l'amélioration du simulateur présenté plus haut dans la partie 3 par l'introduction d'erreurs de transmission basées sur cette série de mesures. Cette étude sera énoncée dans les perspectives.

#### 4.3.1.2 Évaluation du temps de traitement d'un paquet

Le but de cette seconde série de mesures est de quantifier le temps nécessaire au traitement complet d'un paquet par la pile SMAC, c'est-à-dire la durée entre le moment où la demande d'émission est effectuée sur un premier module, jusqu'à l'instant où les données ont été reçues et sont disponibles dans la mémoire d'un second module. Nous avons vu plus haut que SMAC utilise le mode *packet* pour la communication entre le microcontrôleur et le modem radio ; en plus du temps de sérialisation important dû au faible débit sur le médium radio, le temps de transfert d'un paquet de données de la mémoire vers le modem par le bus SPI est a priori non négligeable, comme indiqué sur la figure 4.12. Le délai de transmission global doit être quantifié.

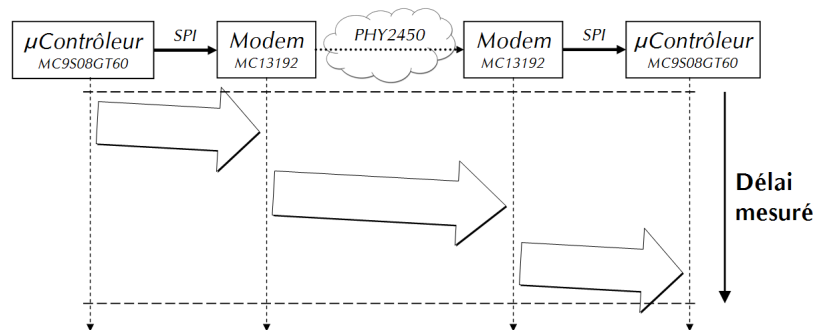


FIG. 4.12 – Illustration des délais engendrés sur toute la transmission avec SMAC et le mode *packet*

L'intérêt de cette série de mesures est triple :

1. évaluer la latence de transmission d'un message quelconque, et plus particulièrement celle d'un *beacon* pour affiner la procédure de synchronisation des noeuds,
2. évaluer la latence minimale et le débit utile maximal pour une transmission utilisant nos transceivers,
3. par extrapolation, évaluer la capacité d'un slot, c'est à dire le volume de données maximum transmissibles pendant cette portion temporelle.

Pour réaliser cette série de mesures, nous avons mis en œuvre deux noeuds basés sur la pile SMAC. Le premier noeud est programmé pour envoyer des trames de longueurs différentes<sup>9</sup> toutes les 10 ms. Les trames sont émises directement, sans méthode d'accès, comme elles le seraient dans un GTS. Juste avant

<sup>9</sup>C'est à dire ici une variation du nombre d'octets utiles, soit, puisque nous sommes au niveau *physique*, une variation de la longueur du PSDU.

la transmission d'une nouvelle trame, le nœud émetteur inverse l'une de ses GPIO. Le second nœud est, quant à lui, bloqué en réception ; à chaque trame reçue, il inverse lui aussi l'une de ses GPIO. Grâce à un analyseur d'état logique temporisé, la différence de temps est mesurée et stockée. Les résultats de cette série de mesures sont représentés figure 4.13.

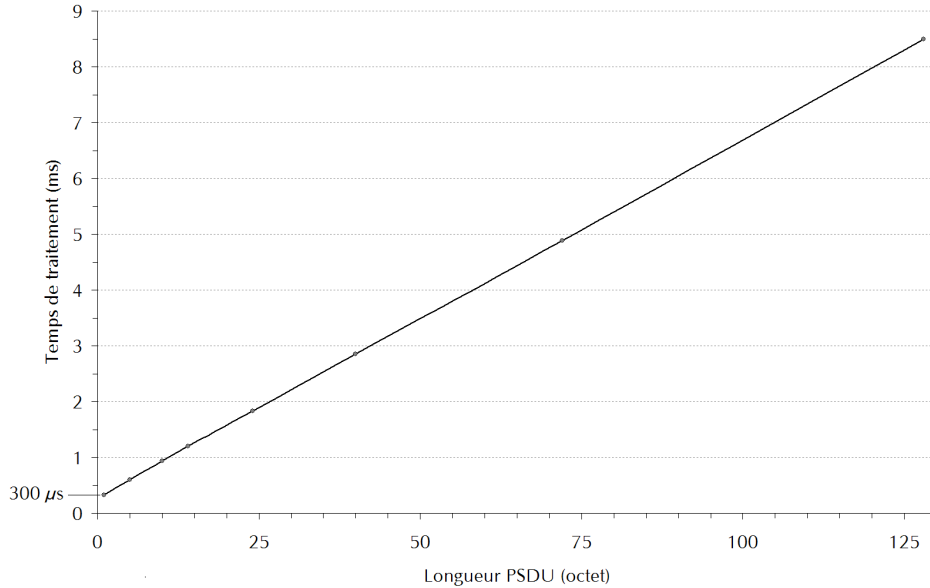


FIG. 4.13 – Mesure du délai total (temps de traitement et de transmission) en fonction de la longueur utile du paquet transmis (PSDU)

La figure 4.13 nous indique que quelque soit le volume de données transmis, un délai d'environ  $300 \mu\text{s}$  est constaté, auquel viennent s'ajouter  $64 \mu\text{s}$  par octet transmis. Le temps de traitement  $T_{\text{traitement}}$  (en ms) d'un paquet de longueur  $l$  (en octets) avec SMAC peut donc être modélisé par l'expression mathématique suivante<sup>10</sup> :

$$T_{\text{traitement}} = \frac{l \cdot 8}{125} + 0,3$$

A priori, le délai initial d'environ  $300 \mu\text{s}$  est inévitable car dû au préambule et à l'en-tête PHY de 802.15.4 qui ont une longueur cumulée de 56 bits, soit  $224 \mu\text{s}$  à  $250 \text{ kbits/s}$ . En revanche, la pente de la droite de la figure 4.13 est de  $64 \mu\text{s}$  par octet, alors qu'à  $250 \text{ kbits/s}$ , le temps de sérialisation d'un octet est de  $32 \mu\text{s}$ , et non  $64 \mu\text{s}$ . Ce temps excessif est probablement dû au transfert de données par le bus SPI qui doit avoir reçu intégralement un paquet avant de pouvoir l'émettre sur la radio (côté émetteur) où de l'envoyer au microcontrôleur (côté récepteur) comme l'illustre la figure 4.12. La même série de mesures devra être effectuée en mode *stream* pour confirmer cette hypothèse mais, dans le cadre du prototype présenté – basé sur SMAC, et donc, sur le mode *packet* – nous considérerons ces performances mesurées.

#### 4.3.1.3 Débit utile maximum

La série de mesures présentée dans la section précédente montre que les performances de SMAC en terme de débit sont relativement faibles. Cependant, compte tenu du temps du traitement d'un paquet, il est possible, par extrapolation, de déduire le nombre maximum de paquets traitables par seconde, soit le débit utile maximum, en fonction de la longueur du paquet (si les paquets sont envoyés les uns à la suite des autres, toujours sans méthode d'accès et sans acquittement). Cette évaluation est représentée

<sup>10</sup>Cette formule sera utilisée plus bas pour calibrer la synchronisation du réseau par les *beacons* dans le mode *suivi de beacon*. Nous aborderons ce point plus loin dans la section 4.3.1.4.

par le graphe de la figure 4.14. Comme nous pouvons le voir sur ce graphe, le débit utile maximum d’une transmission réalisée avec notre dispositif basé sur SMAC est de 120 kbits/s, soit un peu moins de la moitié du débit bande de base.

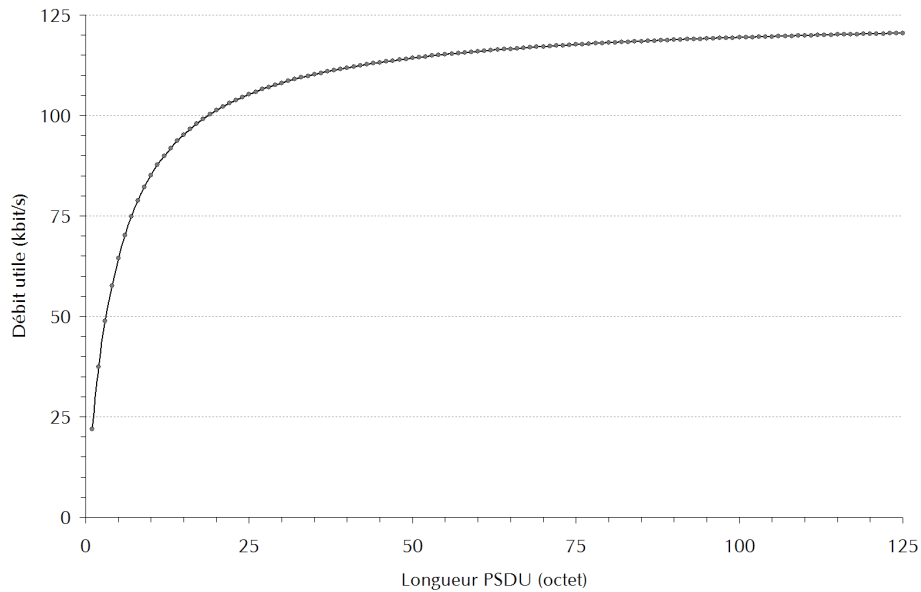


FIG. 4.14 – Évaluation du débit utile maximal en fonction de la longueur du paquet transmis

#### 4.3.1.4 Évaluation de la qualité de la synchronisation entre un coordinateur et ses nœuds

Toujours dans le but d’évaluer la qualité de notre pile prototype, nous avons cherché à évaluer la qualité de la synchronisation des nœuds. En effet, dans le cadre de notre proposition, les éléments de type  *fils*  (cf. section 2.2.2 de ce chapitre) se synchronisent sur les éléments de type  *père*  à la réception des trames  *beacons* . La synchronisation permet ainsi le découpage du temps en slots temporels qui sont ensuite alloués aux différents éléments du réseau pour les accès déterministes. Ce découpage temporel exige donc une excellente synchronisation des nœuds sur leur coordinateur car si la synchronisation est médiocre, les performances du réseau seront globalement moins bonnes :

- les slots ne sont pas synchrones pour tous les nœuds : apparition d’un risque de collisions en bordure des GTS,
- le suivi de  *beacon*  n’est pas optimum : consommation d’énergie accrue.

Pour notre pile prototype, nous avons implémenté le mode suivi de  *beacons*  en étant confronté au problème du temps de traitement variable en fonction de la taille du paquet transmis, comme cela a été évalué dans la section 4.3.1.2 de ce chapitre. Pour simplifier la réalisation de la synchronisation, nous avons implémenté une diffusion de  *beacon*  de taille constante (30 octets), bien que la norme 802.15.4 prévoit que les  *beacons*  puissent être de longueur variable. Nous avons ensuite calibré le  *timer*  utilisé pour le suivi de  *beacons*  grâce à la formule établie dans la section 4.3.1.2. Une fois cette partie du prototype développée, nous avons cherché à quantifier la gigue résiduelle, c’est à dire l’imperfection de la synchronisation.

Tout d’abord, la gigue a été mesurée entre deux nœuds d’une même étoile, comme l’illustre la figure 4.16 : un même coordinateur envoie des  *beacons*  à deux de ses propre nœuds N1 et N2, lui-même étant synchronisé par les  *superbeacons*  qu’il reçoit du supercoordinateur. La resynchronisation des nœuds se fait donc à chaque réception de  *beacon* , soit toutes les 122 ms ( $BO = 3$ ). La figure 4.16 illustre les résultats typiques obtenus pour cette série de mesures. Sur cette figure, l’axe des abscisses représente la



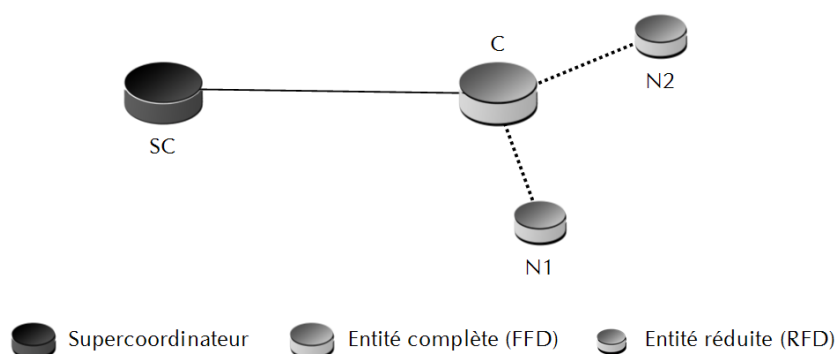


FIG. 4.15 – Topologie pour la mesure de la gigue de synchronisation pour deux nœuds d’une même étoile

différence de synchronisation d’un nœud par rapport à l’autre. L’axe des ordonnées représente le nombre d’occurrences des mesures pour la différence donnée en abscisses. Grâce à cette série de mesures, nous pouvons constater que la synchronisation est très correcte si l’on ramène les valeurs mesurées au temps bit sur le médium de ( $4 \mu s$ ) et à la longueur du préambule (40 bits) nécessaire à la synchronisation du récepteur : le temps moyen (sur l’ensemble des mesures) est de 160 ns, 95% des occurrences sont inférieures à  $16 \mu s$  et la gigue est uniformément répartie (tant en avance qu’en retard d’un nœud à l’autre). Parmi les 5% des occurrences qui expriment une différence de synchronisation supérieure à  $16 \mu s$  (l’aire noire sur la figure 4.16), la plus forte valeur mesurée est  $27 \mu s$  (soit près de 7 temps bit à 250 kbits/s).

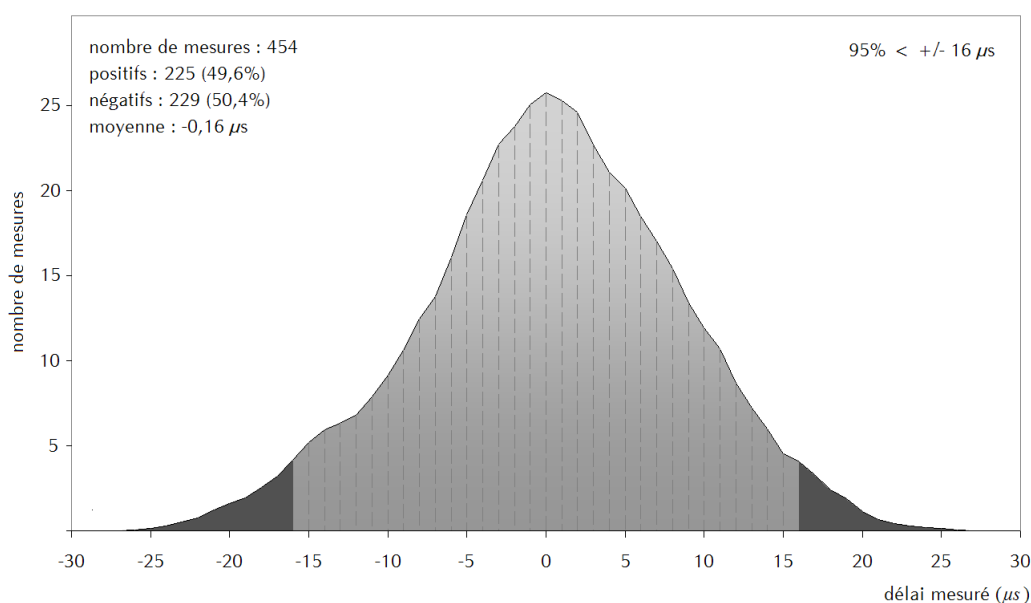


FIG. 4.16 – Quantification de la gigue de synchronisation pour deux nœuds d’une même étoile

#### 4.3.1.5 Conclusion

Ces premières mesures relatives au matériel et à la pile SMAC montrent que cette solution de programmation n’est pas parfaite mais présente l’avantage d’être simple et flexible. Il est tout à fait possible, pour le programmeur, d’intervenir au niveau MAC dans des conditions intéressantes. De plus, les défauts étant maintenant identifiés, le développement de la pile prototype peut être abordé avec sérénité. Les résultats qui suivent portent spécifiquement sur nos propositions concernant la méthode d’accès totalement déterministe.

### 4.3.2 Validation de notre proposition par prototypage

Cette partie présente les études qui ont été menées sur le réseau prototype utilisant la méthode d'accès déterministe, les points validés ici constituant une originalité par rapport au réseau 802.15.4 existant. Dans cette partie, on s'attachera à mettre en évidence la qualité de la synchronisation à plusieurs niveaux, le débit offert maximal et la faisabilité des SGTS.

#### 4.3.2.1 Évaluation de la qualité de la synchronisation entre le supercoordonateur et ses nœuds

Après avoir mesuré la qualité de la synchronisation entre deux nœuds d'une même étoile dans le paragraphe 4.3.1.4, nous avons cherché à quantifier la gigue entre deux nœuds d'étoiles différentes. La topologie du réseau utilisé pour cette mesure est représentée par la figure 4.17. Pour ce nouveau cas, les deux nœuds étudiés sont synchronisés par deux coordinateurs différents, qui sont eux-mêmes synchronisés par le supercoordonateur : nous sommes donc en présence d'une synchronisation relayée (sur deux niveaux), la gigue devrait être théoriquement plus importante puisque la synchronisation des nœuds cumule l'erreur des deux coordinateurs.

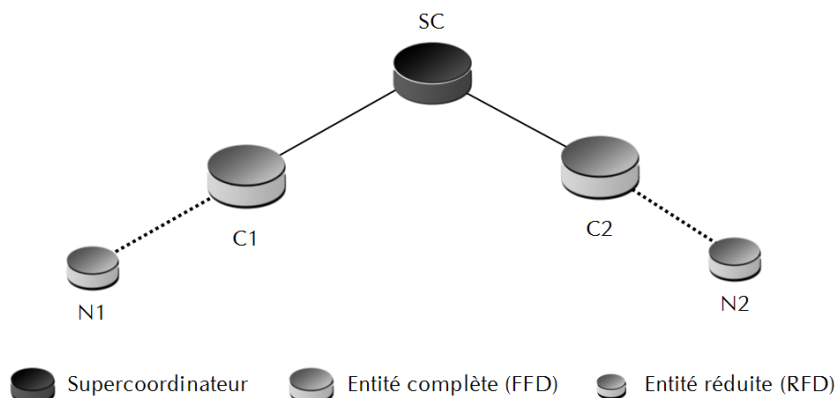


FIG. 4.17 – Topologie pour la mesure de la gigue de synchronisation pour deux nœuds de deux étoiles différentes

La figure 4.18 donne un exemple de résultats obtenus pour cette nouvelle série de mesures. Si on les compare aux résultats concernant la synchronisation d'une étoile, on constate qu'effectivement, la gigue est plus importante mais reste tout de même acceptable puisque 95% des mesures effectuées rapporte une différence de synchronisation inférieure à  $23 \mu s$  et que la plus grande valeur mesurée est  $38 \mu s$ . En revanche, et c'est un détail plus important, la gigue n'est pas uniformément répartie : sur l'ensemble des mesures pratiquées, on constate que l'un des deux nœuds est plus souvent en avance que l'autre. Cette série de mesure révèle probablement un problème mineur dans la procédure de synchronisation qui devra être affinée ; en revanche, nous verrons plus bas que cette asymétrie dans la synchronisation nous a permis de mettre en évidence *l'effet de capture* dans le cadre des SGTS.

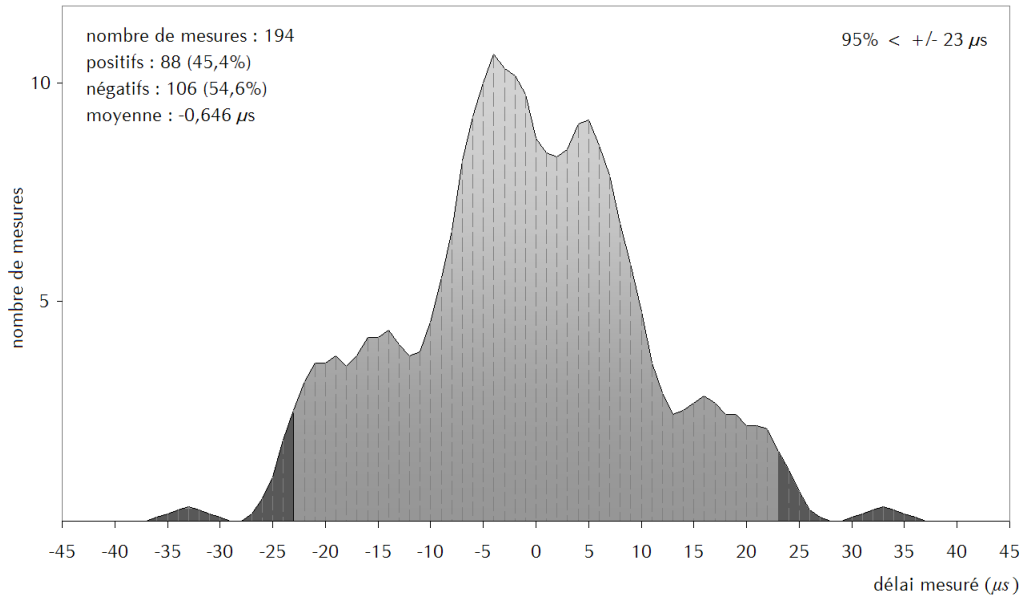


FIG. 4.18 – Quantification de la gigue de synchronisation pour deux nœuds de deux étoiles différentes

#### 4.3.2.2 Capacité maximale d'un slot

Comme nous l'avons vu largement jusqu'ici, notre proposition conserve le concept de *slot* défini par 802.15.4. Il serait donc intéressant, dans le cadre de nos travaux, de déterminer la capacité maximale d'un slot (c'est-à-dire le nombre maximal d'octets, voire le nombre maximal de trames transmissibles dans un slot). Si ce slot est un GTS, il devient alors possible *de quantifier le débit garanti par la méthode d'accès*, c'est-à-dire le volume de données transmises dans un GTS, ramené à la seconde.

Connaissant précisément, pour un nœud de notre réseau prototype, le temps de traitement global d'un paquet<sup>11</sup> (résultats présentés dans le paragraphe 4.3.1.2), il est possible, par extrapolation, d'évaluer la capacité d'un slot en fonction du paramètre  $BO$  et du niveau de réservation  $n$  de ce slot ( $n_{GTS}$ ,  $n_{PDS}$  ou  $n_{GBS}$ ). Cependant, pour évaluer la capacité d'un slot, il est nécessaire de considérer deux cas, avec ou sans acquittement contenu dans le slot<sup>12</sup>. Dans le premier cas, il est nécessaire de considérer le temps de traitement de la trame d'acquittement en plus du temps de traitement de la trame envoyée. Nous avons vu dans le chapitre 2, section 2.3.3.4 (page 68) que la trame d'acquittement 802.15.4 a une longueur de 5 octets, soit, d'après les mesures réalisées plus haut, un temps de traitement de 620  $\mu\text{s}$ . Nous avons vu également dans le chapitre 3, section 2.3.5 (page 95) qu'un temps intertrame de 192  $\mu\text{s}$  devait être respecté. Connaissant toutes ces données temporelles, il est possible de déterminer la capacité maximale d'un slot avec ou sans acquittement, en fonction des paramètres  $BO$  et  $n$ . Le graphe de la figure 4.19 illustre ces résultats.

Cette étude nous permet de constater qu'avec notre prototype, il n'est pas possible d'envisager un réseau où  $BO = 0$ . En effet, pour cette valeur, un slot n'a pas la capacité nécessaire pour l'envoi d'une trame, même très petite, et son acquittement. Le  $BO$  minimal, pour notre réseau prototype, sera donc de 1. De plus, nous pouvons constater que la valeur  $BO = 3$  donne des résultats intéressants car elle permet un débit proche du maximum, tout en permettant une durée de supertrame assez courte (122 ms, donc une latence globale faible, comme montré par la simulation). Cette valeur, qui présente un compromis intéressant, a été retenue pour notre réseau prototype, et, par la même occasion, pour les travaux de simulation qui ont été exposés plus haut dans la section 3.1.

<sup>11</sup>c'est-à-dire : envoi au modem, transmission sur l'air puis réception

<sup>12</sup>Si le slot est un GTS, il sera utilisé pour envoyer des trames de données, qui devront être acquittées. Si le slot est un GBS, il n'est pas nécessaire de prévoir ce temps additionnel puisque les *beacons*, comme toutes trames *diffusées*, ne sont pas acquittés.

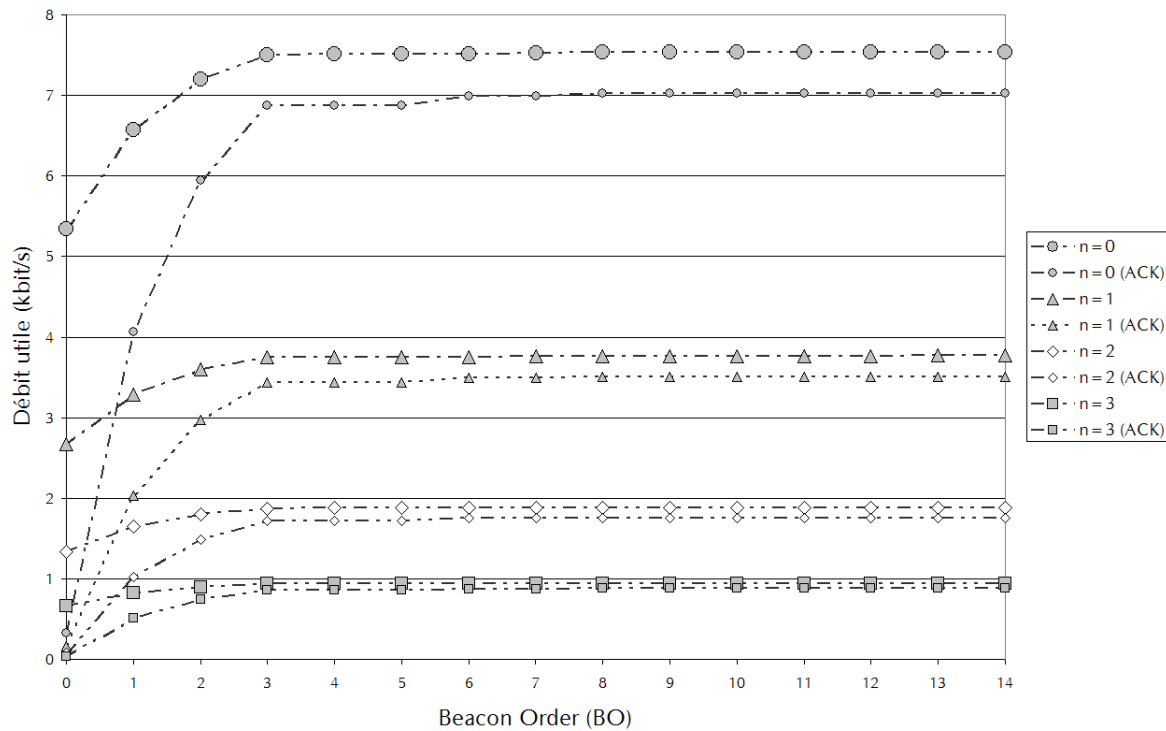


FIG. 4.19 – Évaluation du débit utile maximal pour un slot déterministe, en fonction du niveau de  $BO$  et du niveau de réservation  $n$

Cette étude par extrapolation permet également de comparer la méthode d'accès au médium déterministe sur le plan du débit, avec le CSMA/CA utilisé dans la CAP (cf. l'étude de performances réalisée avec NS2 sur le débit utile maximal en fonction de la charge globale du réseau, partie 3.2.3). A titre de comparaison, si nous affectons un nombre croissant de GTS (sans dépasser la limite  $aMinCAPlength$ , cf. chapitre 2, paragraphe 2.3.1.2), nous pouvons obtenir des résultats qui seront comparables. La figure 4.20 illustre ces résultats, et peut-être comparée aux performances obtenues avec CSMA/CA (figure 4.8, page 126) avec  $BO = 0$ .

Comme nous pouvons le voir sur le graphe de la figure 4.20, si la charge globale du réseau augmente, les performances du réseau ne s'écroulent pas, comme c'est inévitablement le cas pour le CSMA/CA. L'accès au médium déterministe permet bel et bien de garantir une certaine bande passante qui ne dépend alors que du nombre de GTS alloués, de leur fréquence  $n_{GTS}$  et de la périodicité des *beacons*  $BO$ . Nous pouvons également noter que les performances évoquées ici ne tiennent pas compte de la possibilité d'attribuer un GTS simultanément à deux noeuds différents (concept de SGTS, qui sera l'objet du paragraphe suivant); avec cette fonctionnalité, les performances devraient être encore meilleures.

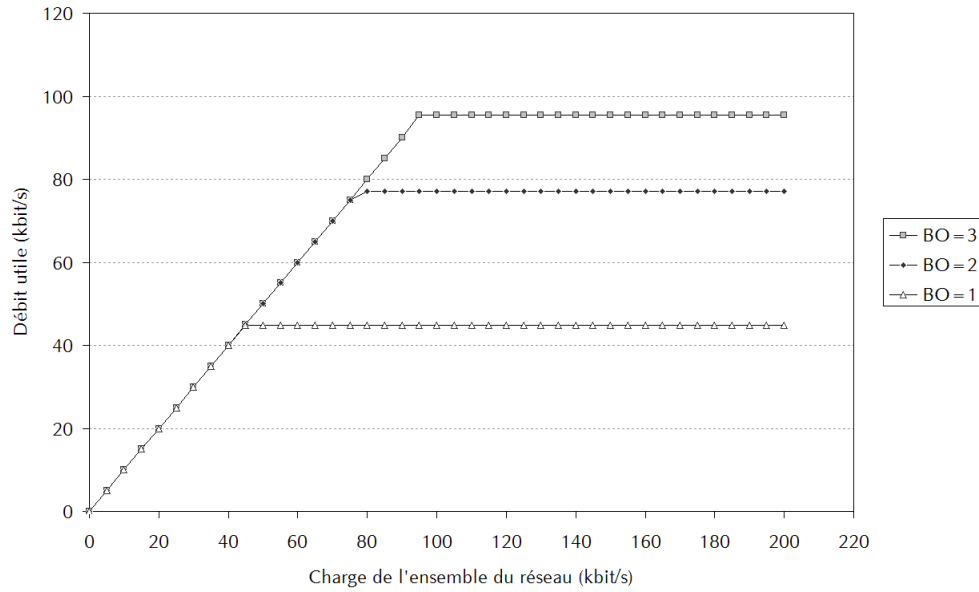


FIG. 4.20 – Évaluation du débit maximum offert en fonction de la charge globale du réseau

#### 4.3.2.3 Validation du concept de SGTS

Dans le chapitre 3, nous avons évoqué la possibilité d'affecter un même slot temporel à deux nœuds différents, ceci dans le but de palier l'éventuelle rareté des slots, tout en conservant l'aspect déterministe de la méthode d'accès. Nous avons désigné ce slot partagé « SGTS », pour *Simultaneous GTS*. La mise en œuvre de cette fonctionnalité repose sur l'hypothèse que deux émetteurs peuvent accéder au médium en même temps – si et seulement si – aucun des récepteurs n'est dérangé par l'émetteur « perturbateur ». Dans la pratique, nous avons limité le problème à deux couples de nœuds émetteurs/récepteurs<sup>13</sup>. Il reste à prouver que cette fonctionnalité peut être réellement mise en œuvre et à trouver les conditions de faisabilité.

Pour réaliser cette étude, nous avons repris la topologie de la figure 4.17, composée de cinq nœuds, tous à portée les uns des autres :

- un supercoordonateur (SC),
- un premier coordonateur (C1),
- un second coordonateur (C2),
- un premier nœud (N1) lié à C1,
- un second nœud (N2) lié à C2.

Afin de déterminer les conditions limites de l'utilisation simultanée du médium, nous avons implémenté le protocole de mesure suivant, dont la succession des messages est illustrée figure 4.21 :

1. chaque nouvelle supertrame débute normalement par un *superbeacon* émis par SC sur le slot 0. Les coordonateurs C1 et C2 se synchronisent,
2. C1 émet son *beacon* sur le slot 1. Le nœud N1 se synchronise,
3. C2 émet son *beacon* sur le slot 2. Le nœud N2 se synchronise,

<sup>13</sup>qui sont forcément dans deux étoiles différentes puisqu'au sein d'une étoile, toutes les communications passent par le coordonateur...

4. N1 émet une trame de données quelconques destinées à C1 dans le slot 3. C1 réceptionne la trame et prend note de la puissance avec laquelle il a reçu le signal venant de son nœud. Si C2 reçoit le signal émis par N1, il note également la puissance du signal reçue,
5. N2 émet une trame de données quelconques destinées à C2 dans le slot 4. Comme C1 dans le slot précédent, C2 réceptionne la trame et prend note de la puissance reçue. Si C1 reçoit le signal émis par N2, il note aussi la puissance reçue,
6. N1 et N2 émettent en même temps, dans le slot 5, une trame destinée à leurs coordinateurs respectifs. Les coordinateurs envoient alors un accusé de réception sur leur port série ainsi que la différence des puissances reçues. Un accusé positif signifie que la trame attendue a été reçue ; un accusé négatif signifie la réception d'un signal incohérent ou de la trame émise par l'autre nœud. Sur un PC, le résultat de chaque mesure est enregistré.
7. pour finir, N1 et N2 utilisent le temps restant avant la prochaine supertrame pour faire varier leur puissance d'émission permettant d'étendre la plage des mesures.

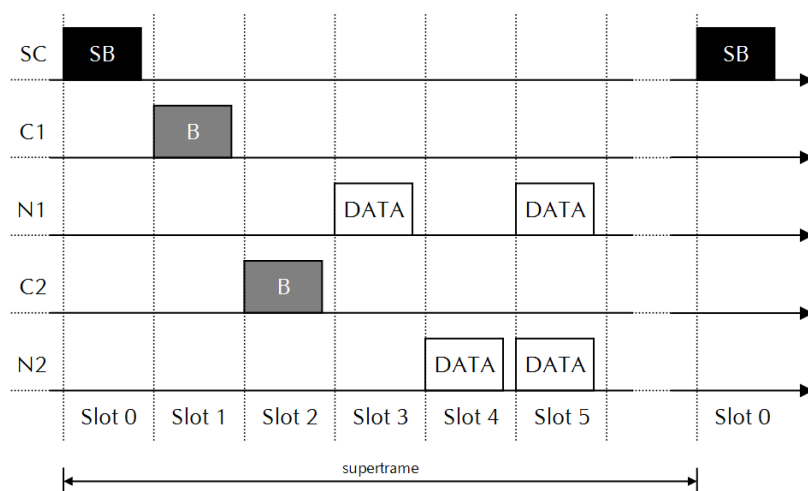


FIG. 4.21 – Succession des messages échangés sur le médium dans le protocole de test pour la faisabilité des SGTS

Cette série de mesures a également été réalisée dans la chambre anéchoïque. Près de 30000 émissions simultanées ont été réalisées. Les résultats sont visibles sur la figure 4.22. Sur ce graphe, l'axe des abscisses représente la différence du signal reçu – une puissance positive signifie que le coordinateur a reçu le signal émis par N1 avec plus de force que celui reçu de N2 – tandis que l'axe des ordonnées représente le taux des trames reçues correctement. Deux courbes sont représentées : une première pour le coordinateur C1 cherchant à recevoir le message de N1 et une seconde pour le coordinateur C2 cherchant à recevoir le message de N2.

Les résultats de cette série de mesures en l'absence de bruit prouvent que le principe de SGTS est tout à fait envisageable dans les conditions qui ont été celles de la mesure : dans la situation des tests, une zone d'environ 10 dB seulement doit être évitée ; en dehors de cette zone, on peut considérer que deux accès simultanés sur le médium ne perturbent pas la bonne réception des deux récepteurs. Ces résultats devront certes être confrontés à une situation réelle (hors de la chambre anéchoïque) dans laquelle le bruit électromagnétique peut être important : la marge de 10 dB devra alors être augmentée, si possible de façon adaptative en fonction du rapport *signal sur bruit*.

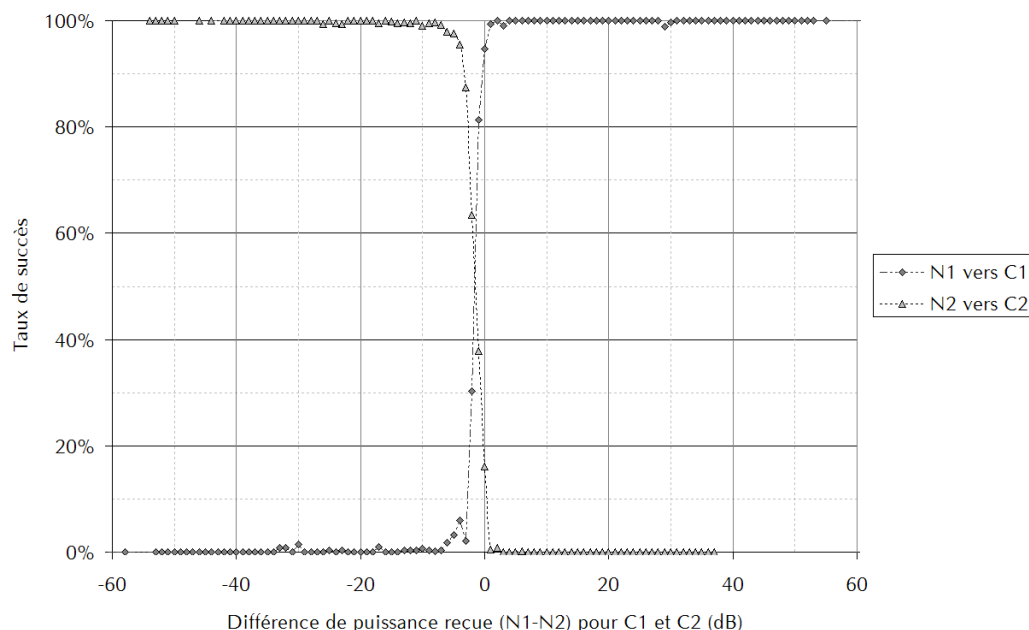


FIG. 4.22 – Taux de trames réceptionnées correctement en fonction du rapport de puissances reçues

Cependant, il est intéressant de constater que les deux courbes de la figure 4.22 sont « légèrement décalées vers la gauche » par rapport à ce qui était naturellement attendu ; en effet, les deux courbes se croisent à une valeur de -3 dB, ce qui est surprenant. Deux hypothèses ont été avancées :

- soit la mesure manque de précision (imprécision de l'outil de mesure, mauvaise manipulation ou mauvaises conditions),
- soit la mesure est très précise et relève un effet intéressant qui permet à l'un des nœuds d'être favorisé par rapport à l'autre : il est reçu correctement par le récepteur, même si l'autre émetteur est reçu avec un signal plus puissant (jusqu'à 3 dB plus fort).

Nous avons refait de nombreuses mesures, mais le phénomène constaté était toujours présent. Nous avons donc supposé que nous mesurions un phénomène réel et non un artefact. Nous avons alors repris l'étude de la synchronisation présentée plus haut dans la section 4.3.1.4 et constaté que le nœud favorisé dans le cadre de cette étude (N1) est celui qui dispose, en moyenne, d'une légère avance dans le dispositif de synchronisation. A ce stade, l'étude nous permettrait d'avancer l'hypothèse suivante : si l'on souhaite émettre deux trames en même temps (contention), et que l'une est en réalité légèrement en avance par rapport à l'autre, la première transmise a le plus de chance d'être reçue correctement, même si la seconde est reçue avec plus de puissance (jusqu'à 3 dB plus fort). Il semblerait que nous ayons mis en évidence un *effet de capture* : les émetteurs n'émettant pas strictement sur la même fréquence, en cas d'émission concurrentes, le récepteur reste synchronisé sur l'émetteur qui a commencé à émettre en premier.

Afin de confirmer cette hypothèse, nous avons repris le réseau prototype en modifiant le protocole de manière à ce que les deux nœuds en concurrence soient synchronisés sur le même coordinateur. Les résultats de cette série de mesures sont présentés sur le graphe de la figure 4.23 et confirment que, dans le cas d'une synchronisation ne favorisant aucun des deux nœuds, la simultanéité de l'accès au médium ne favorise aucun des nœuds.

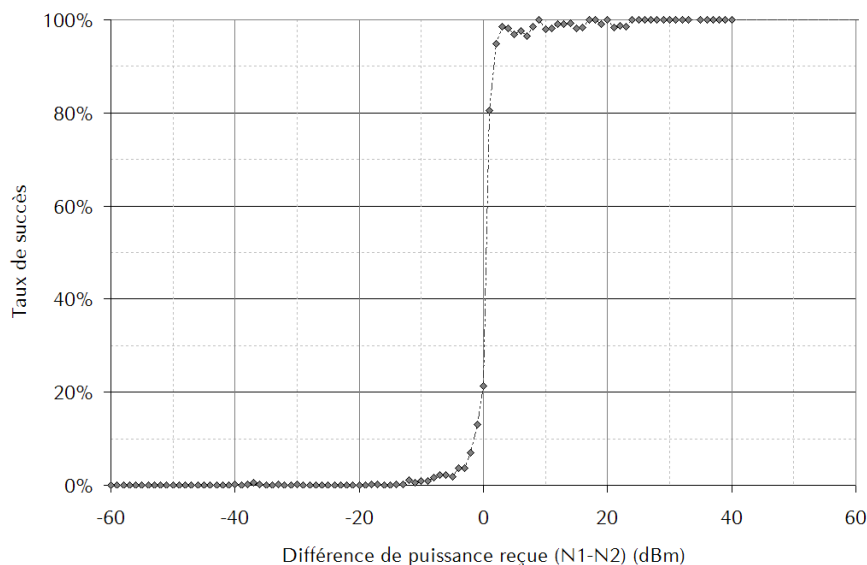


FIG. 4.23 – Taux de trames réceptionnées correctement en fonction du rapport de puissances reçues

### 4.3.3 Conclusion sur les mesures réalisées sur le prototype

Les différentes études réalisées sur le réseau prototype utilisant la méthode d'accès déterministe donnent des résultats encourageants :

- Les concepts de *synchronisation à plusieurs niveaux* et *SGTS*, difficilement validables par simulation, et encore moins par Réseaux de Petri, ont été validés avec succès grâce au prototypage. Nous avons pu constater à cette occasion que les SGTS sont tout à fait implémentables en tenant compte d'une marge relativement faible. Nous notons également que la synchronisation par *cascade de beacons* augmente considérablement la gigue de synchronisation. Une structure étendue à  $n$  niveaux, qui sera évoquée en perspectives de cette thèse, risque donc d'introduire des difficultés à ce niveau.
- La série de mesures sur le temps de traitement d'un paquet nous a permis, par extrapolation, d'établir les performances de notre méthode d'accès en terme de débit, et ce, en nous basant sur des mesures issues du prototypage. Grâce à cette évaluation, il est possible d'établir des abaques liant les paramètres temporels  $BO$  et  $n_{GTS}$  avec le débit minimum garanti par la couche MAC.
- Enfin, les autres mesures ont permis d'approcher finement les performances liées à la partie radio du composant FREESCALE utilisé pour le réseau prototype (caractérisation de la sensibilité du récepteur, taux de trame perdues typique, etc.).

## 4.4 Applications de démonstration

Comme cela a été évoqué plus haut, l'aspect prototypage de la thèse a été développé dans le but de valider certaines propriétés de la proposition, mais aussi par intérêt pédagogique et démonstratif. Le prototypage et la manipulation du matériel IEEE 802.15.4 / ZigBee a donc donné lieu à de nombreux projets parallèlement à ces travaux de recherche. Ces projets n'ont, bien sûr, pas un intérêt direct à une quelconque contribution de recherche mais ils s'inscrivent dans un cursus de formation (stages, projets tuteurés) et il nous paraît important de les évoquer ici.



#### 4.4.1 Capteur de ligne ZigBee

Le premier projet a été mené au laboratoire ICARE au cours de l'été 2005 par Mathieu GERBE, un étudiant de seconde année DUT GTR (désormais DUT *Réseaux et Télécoms*) de l'IUT de Clermont Ferrand. Le but du stage [GERB 05] était de mettre en œuvre un robot EVOLUTION ROBOTICS *ER1* [EVOL] et de l'équiper de deux noeuds IEEE 802.15.4, constituant ainsi une plateforme opérationnelle robotique de test et de développement. Ce projet est également lié à une collaboration de recherche entre le LIMOS (*Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes*) et ICARE sur le projet de recherche Wi-Bot [VDB3 05].

Dans ce projet, l'unité de commande du robot (PC portable) était dotée d'un nœud faisant office de coordinateur. Nous avons également développé un capteur optique dont le but était l'identification d'une ligne au sol ; ce capteur était lui aussi pourvu d'un nœud ZigBee pour communiquer ses informations au coordinateur. La topologie ainsi créée était la topologie interne au robot évoquée dans le début du chapitre 3. La figure 4.24 représente le robot et son capteur.



FIG. 4.24 – Le robot ER1, le capteur infrarouge et le coordinateur (module ZigBee ICARE)

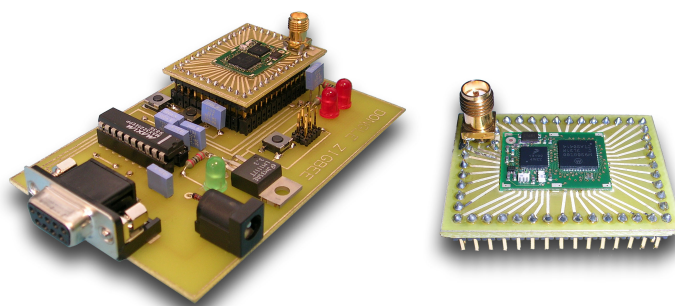


FIG. 4.25 – Réalisations de modules ZigBee ICARE

Ce premier projet a également permis de réaliser plusieurs cartes ZigBee basées sur des composants FREESCALE. Dans le but de pouvoir réutiliser facilement les modules ZigBee pour une autre application,

une solution « carte mère / carte fille » a été développée. Deux versions de carte mère ont été réalisées : une pour les capteurs (nœud terminal) et une avec un port RS232 (coordinateur). De même, deux versions de carte fille ont été réalisées : une première avec antenne imprimée et une seconde avec un connecteur (type SMA) pour l'utilisation d'une antenne externe. Ces réalisations sont représentées sur la figure 4.25 et ont également servi pour la phase de prototypage de la méthode d'accès déterministe de notre thèse. Une documentation technique [VDB2 05] a été rédigée à la suite de la réalisation de ces cartes.

#### 4.4.2 Passerelle IP/ZigBee

Un projet de passerelle IP/ZigBee a été confié durant l'année 2006/2007 à des étudiants de seconde année DUT R&T de l'IUT de Blagnac dans le cadre des projets tuteurés. Le but était le développement d'un dispositif autonome et indépendant de tout ordinateur pour permettre l'échange de données entre les deux réseaux, et ce, à un niveau applicatif. La passerelle IP/ZigBee sera ensuite intégrée dans le réseau de robots pour permettre la télécommande et le *monitoring* des appareils mobiles.

La partie matérielle, constituée d'une carte WiFi *Compact Flash* (et de son support implémentant une pile IP et un port RS232) et d'un module ZigBee ICARE, a été réalisée par les étudiants [BETH 07]. La figure 4.26 représente la carte réalisée. La partie logicielle fera l'objet d'un projet tuteuré l'année prochaine.

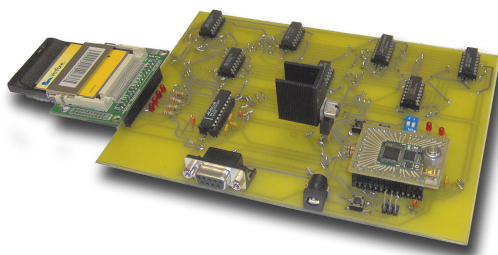


FIG. 4.26 – Passerelle IP/ZigBee

#### 4.4.3 Flotte de robots ZigBee

Le dernier projet autour de 802.15.4/ZigBee a été proposé au printemps 2007 à des élèves-ingénieurs RT de l'INSA de Toulouse suite au rapprochement du LESIA et d'ICARE dans le LATTIS<sup>◇</sup>. Le thème était directement dans la continuité de celui du capteur de ligne ZigBee : il s'agissait d'équiper une flotte de plusieurs robots EVOLUTION ROBOTICS *ER1* de nœuds ZigBee (un coordinateur et un à deux capteurs par robot) et, par là même, de constituer une plateforme opérationnelle pour le déploiement futur d'un réseau à Qualité de Service (grâce à la méthode d'accès déterministe) appliqué à une flotte de robots mobiles. Les deux robots évoluaient sur un même parcours en suivant une ligne au sol comme illustré figure 4.27. L'un des robots était équipé d'un capteur d'obstacle (sonar ultrason) qui lui permettait de s'arrêter si un obstacle était détecté, et de demander à l'autre robot, via ZigBee, de s'arrêter également. D'un point de vue réseau, ces capteurs constituaient des sources d'informations critiques sur le plan temporel, répondant ainsi à nos exigences quant à la mise en œuvre d'une plateforme applicative de test pour notre méthode d'accès déterministe.

Ce projet inclut un niveau *réseau* (au sens OSI) plus abouti dans la mesure où les communications entre les robots nécessitent la mise en œuvre d'un protocole de routage. A ce titre, la couche *réseau* de ZigBee incluant AODV, a été implémentée et testée par les étudiants. Le projet est en cours de finalisation.

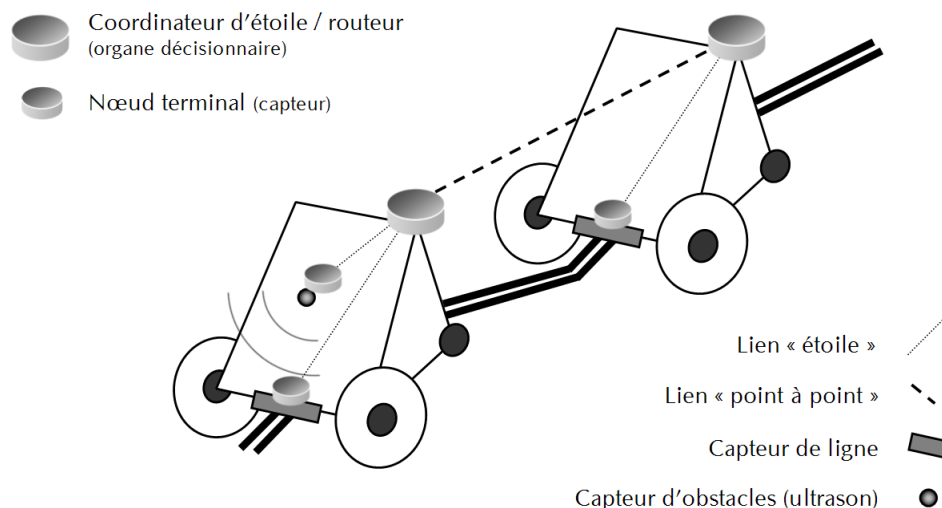


FIG. 4.27 – Les deux robots suiveurs de ligne et le réseau formé

#### 4.4.4 Bilan sur les applications de démonstration

Les projets réalisés autour de 802.15.4 et ZigBee, outre l'aspect pédagogique, nous ont permis de réaliser des plateformes matérielles que nous avons pu utiliser, par la suite, comme nœuds du réseau prototypé. De plus, les sujets proposés ont permis l'ouverture vers des domaines d'applications autres que l'unique champ de la robotique présenté en début de chapitre 3. De nouvelles applications telles que la surveillance d'environnements à risque (bateaux, centrales), nécessitant un système de communication industriel sans fil à fortes contraintes temporelles et à QoS, sont dès lors envisageables et seront mises en œuvre dans un projet de bien plus grande envergure avec plusieurs partenaires universitaires et industriels sur le plan national.



# Bibliographie du chapitre 4

- [ATAM 94] Y. ATAMNA – *Réseaux de Petri temporisés stochastiques et bien formés : définition, analyse et application aux systèmes distribués temps réel* – Doctorat de l'Université Paul Sabatier, Toulouse, France (1994)
  - [BERT 04] B. BERTHOMIEU, P.O. RIBET et F. VERNADAT – *The tool TINA - Construction of Abstract State Spaces for Petri Nets and Time Petri Nets* – International Journal of Production Research, Vol. 42, no 14, (Juillet 2004)
  - [BETH 07] J. BETHMONT et P. POMMIER – *Réalisation d'une passerelle IP/ZigBee* – Rapport de projet tuteuré, équipe ICARE, Blagnac (Mars 2007)
  - [BRAM 83] G.W. BRAMS – *Réseaux de Petri : théorie et pratique* – Masson, Paris (1983)
  - [EVOL] Le site web de EVOLUTION ROBOTICS – [http://www.evolution.com/er1/what\\_overview.masn](http://www.evolution.com/er1/what_overview.masn)
  - [FLOR 85] G. FLORIN – *Réseaux de Petri stochastiques : théorie et méthode de calcul* – Doctorat de l'Université Pierre et Marie Curie Paris VI, Paris, France (1985)
  - [GANG 04] Gang LU, Bhaskar KRISHNAMACHARI et Cauligi S. RAGHAVENDRA – *Performance evaluation of the IEEE 802.15.4 MAC for Low-Rate Low-Power Wireless Networks* – Workshop on Energy-Efficient Wireless Communications and Networks (EWCN '04), IEEE International Performance Computing and Communications Conference (IPCCC), Phoenix, Arizona, USA (Avril 2004)
  - [GERB 05] M. GERBE – *Réalisation d'un capteur sans fil communicant par ZigBee* – Rapport de stage, équipe ICARE, Blagnac (Juin 2005)
  - [HPSI] Le site web du logiciel d'animation de RdP HPSIM, de H. ANSCHUETZ – <http://www.winpesim.de>
  - [IEE3 03] IEEE COMPUTER SOCIETY – *IEEE 802.15.4/D18, Draft Standard : Low Rate Wireless Personal Area Networks* – (Février 2003)
  - [KHOU 06] T. KHOUTAIF et G. JUANOLE – *Formal modelling and evaluation of the data transfer phase of the ACL links on the WPAN Bluetooth* – 11th International Conference on Emerging Technologies and Factory Automation (ETFA'2006), Prague, République Tchèque (Septembre 2006)
  - [LAAS] Le site web du LAAS – <http://www.laas.fr>
  - [MARS 85] M.A. MARSAN, G. BALBO, A. BOBBIO, G. CHIOLA, G. CONTE et A. CUMANI – *On Petri Nets with Stochastic Timing* – International Workshop on Timed Petri Nets (INTPN'85), Turin, Italie (Juillet 1985)
  - [MAUR 05] R. MAURICE – *Contribution à la méthodologie de conception système : application à la réalisation d'un microsystème multicapteurs communicant pour le génie civil* – Doctorat de l'Institut National Polytechnique de Toulouse, Rapport LAAS n°05646, Toulouse, France (2005)
  - [MERC 06] J.J. MERCIER – *Computers II : Séquence après séquence, Introduction aux Gracets* – chapitre 3.2, Collection Technosup, Ellipses (2006)
  - [MERL 76] P.M. MERLIN – *Recoverability of communication protocols, implication of a theoretical study* – IEEE Transaction on Communications, 24(9) :1036-1043 (1976)
  - [NS] Le site web de Network Simulator (NS) – <http://www.isi.edu/nsnam/ns>
  - [OLC] Le site web du groupe OLC – <http://www2.laas.fr/laas/1-4286-Groupe-OLC.php>
-

- [OPNE] Le site web d'OPNET – <http://www.opnet.com>
- [PETR 06] M. PETROVA, J. RIIHIJARVI, P. MAHONEN et S. LABELLA – *Performance Study of IEEE 802.15.4 Using Measurements and Simulations* – *IEEE Wireless Communications and Networking Conference* (WCNC'06), vol. 1, pp : 487- 492, Las Vegas, Nevada, USA (Avril 2006)
- [QUAL 02] SCALABLE NETWORK TECHNOLOGIES – *Speed & Scalability : Requirements for Industrial Strength Network Simulators* – QUALNET White Paper (2002)
- [QUAL] Le site web de Qualnet – <http://www.scalable-networks.com>
- [SAMS] SAMSUNG/CUNY – *NS2 simulator for 802.15.4* – <http://www-ee.ccny.cuny.edu/zheng/pub>
- [TINA] Le site web de TINA – <http://www.laas.fr/tina>
- [VDB2 05] A. VAN DEN BOSSCHE et M. GERBE – *Les modules IEEE 802.15.4 ICARE* – Documentation technique illustrée, document interne ICARE (Juin 2005)
- [VDB3 05] A. VAN DEN BOSSCHE – *IEEE 802.15.4 / ZigBee : un réseau sans fil à faible consommation énergétique* – 2<sup>èmes</sup> rencontres des Sciences et Technologies de l'Information (ASTI'2005), Aubière, France (Octobre 2005)
- [VINT 95] *The NS Manual (formerly NS Notes and Documentation)* par les contributeurs du projet VINT – UC Berkeley, LBL, USC/ISI, and Xerox PARC (1995)
- [ZHEN 04] J. ZHENG et M. J. LEE – *Will IEEE 802.15.4 make ubiquitous networking a reality? : A discussion on a potential low power, low bit rate standard* – *IEEE Communications Magazine*, vol. 27, no. 6, pp. 23-29 (2004)
- [ZHEN 06] J. ZHENG et M. J. LEE – *A comprehensive performance study of IEEE 802.15.4* – *Sensor Network Operations*, IEEE Press, Wiley Interscience, chapitre 4, pp. 218-237 (2006).

# Conclusion et perspectives

## Conclusion générale

Aujourd'hui, la grande majorité des technologies de WLAN/WPAN présentes sur le marché ne permet pas de mettre en œuvre des réseaux adaptés aux transports d'informations à fortes contraintes temporelles. La plupart des normes de réseaux sans fil traitent la problématique de la méthode d'accès au médium en se basant sur des délais aléatoires qu'il est impossible de borner. De plus, quelque soit la technologie de transmission mise en œuvre (et même dans des conditions quasi idéales), la couche physique est imparfaite car elle introduit des erreurs qui doivent être corrigées par les couches supérieures. Dans le cadre d'un réseau transportant des informations contraintes temporellement, ces erreurs peuvent engendrer des délais de transmission importants et retarder l'acheminement d'un message. Il est donc nécessaire de pouvoir les borner et travailler avec des méthodes et des protocoles qui n'introduisent pas, en plus, des phénomènes aléatoires. Afin de limiter fortement ces délais, il nous a paru nécessaire de considérer que l'incertitude temporelle cumulée introduite par chaque couche doit être négligeable devant l'incertitude introduite par la couche physique qui, elle, restera pour le moment, imprévisible et non fiable.

Dans le premier chapitre sur l'état de l'art des technologies de réseaux sans fil et de Qualité de Service, nous avons présenté plusieurs normes et protocoles existants. Nous avons conclu sur la réelle nécessité d'introduire des fonctionnalités de QoS au niveau le plus bas de la pile protocolaire pour pouvoir assurer la Qualité de Service globale sur tout le réseau. Sans une gestion fine de la QoS à chaque niveau de la pile OSI, l'accumulation des incertitudes aboutit à un réseau globalement non fiable sur le plan temporel et dont le comportement est difficilement prévisible. En particulier, l'un des principes fondamentaux de la Qualité de Service est le *contrôle de l'admission* ; or ce dispositif nécessite une connaissance très juste de la capacité du réseau à tout instant et il est donc important que les couches les plus basses informent les processus gérant la QoS de cette capacité, si possible en temps réel. De plus, considérant que le médium imparfait constitue l'élément le moins fiable de l'ensemble des constituants du réseau, il est alors très regrettable de constater qu'un message reçu correctement au niveau de la couche physique soit détruit par une couche supérieure par exemple parce que le réseau est saturé. Dans une pile protocolaire, au dessus de la couche *physique*, le premier élément susceptible d'introduire des incertitudes est l'algorithme de la méthode d'accès au médium. Dans le but de réduire les incertitudes provoquées par les couches 2 et supérieures, nous avons été amené à nous rapprocher des méthodes d'accès au médium présentant un caractère non aléatoire. Plusieurs technologies présentent de telles possibilités : HiperLAN/2, 802.15.1, 802.11 PCF, 802.11e, 802.15.4 GTS, mais tous ces algorithmes reposent tout de même sur des demandes elles-mêmes effectuées en mode aléatoire, ce qui ne permet pas de garantir à 100% les temps de réponse, par exemple si le réseau est saturé. Notre proposition a donc tenté de proposer une solution à ce problème.

Dans le second chapitre, nous avons présenté les normes et spécifications des technologies de réseaux personnels sans fil IEEE 802.15.4 et ZigBee sur lesquelles se sont basés nos travaux. Outre l'aspect novateur de cette technologie – la sortie des premiers modules a coïncidé avec le début de nos travaux – les possibilités offertes de reprogrammation totale de la couche MAC nous ont permis d'entrevoir des possibilités de prototypage et, donc, des résultats de performances réels. De plus, nos travaux étant

destinés à des applications de réseaux sans fil industriels autonomes, l'aspect faible consommation de cette technologie nous a d'autant plus encouragé. De plus, l'étude de cette technologie nous a permis d'identifier un certain nombre de faiblesses pour des applications de transport d'informations à fortes contraintes temporelles. En effet, comme d'autres technologies présentées dans le premier chapitre, IEEE 802.15.4 propose une méthode d'accès avec contention basée sur CSMA/CA et une méthode d'accès sans contention basée sur le principe de slots réservés (GTS). Si les GTS permettent à quelques nœuds privilégiés qui en font la demande d'être temporairement épargnés par le phénomène des collisions, le protocole de demande de GTS passe par le mode avec contention, ce qui ne permet pas d'assurer avec certitude que l'obtention d'un ou plusieurs GTS peut se faire dans une période temporelle bornée. Par extension, nous avons relevé qu'aucun mécanisme de gestion du réseau (association, désassociation, demande indirecte de données, demande de renouvellement de GTS, etc.) ne peut se faire de façon déterministe.

Une étude fine de la gestion de l'accès au médium dans le cadre de IEEE 802.15.4 nous a permis de proposer six grandes améliorations dans le protocole, présentées dans le chapitre 3. Ces améliorations sont les suivantes :

1. possibilité de pratiquer des entrées déterministes dans le réseau,
2. création d'un protocole entre coordinateurs pour éviter les collisions de GTS,
3. suppression délimitation entre CAP/CFP,
4. plusieurs niveaux de GTS créés, pour plusieurs classes de services,
5. continuité du service déterministe,
6. possibilité d'allouer deux GTS sur le même slot sans risque de perturbation (SGTS).

Ces propositions ont fait l'objet de plusieurs analyses dont les résultats ont été exposés et commentés dans le quatrième chapitre de cette thèse. Dans le but de pouvoir valider de façon adéquate les différents aspects de nos propositions, nous avons utilisé trois méthodes de validations complémentaires :

- les Réseaux de Pétri pour la validation par des méthodes formelles de l'aspect séquençement de la méthode d'accès,
- la simulation pour l'analyse des performances temporelles dans la phase d'*association déterministe au réseau*,
- le prototypage pour l'évaluation du débit garanti par la méthode d'accès, la qualité de la synchronisation sur *cascade de beacons* ainsi que la validation du principe de SGTS.

Chaque contribution proposée a été traitée par l'une de ces trois méthodes : ces trois études complémentaires nous ont permis d'approcher la phase de validation par trois directions différentes, enrichissant ainsi nos résultats et nos conclusions.

Ainsi, nous avons modélisé l'algorithme proposé pour la méthode d'accès par les Réseaux de Pétri et testé ce modèle avec le logiciel TINA qui propose des fonctionnalités de validation formelle. Nous avons pu constater que notre modèle ne souffrait d'aucun défaut de séquençement ; nous l'avons alors implémenté dans un simulateur logiciel en nous focalisant spécifiquement sur la phase d'*association au réseau*, le simulateur ajoutant l'aspect temporel pour l'obtention des paramètres de performances. Nous avons constaté que la méthode d'accès présente bien le caractère déterministe car toute demande suscitée par un nœud du réseau obtient une réponse en un temps borné (minimum et maximum). Parallèlement aux travaux de simulation, nous avons mené une étude basée sur un réseau prototype implémentant la méthode d'accès déterministe. Nous avons cherché, dans un premier temps, à évaluer les caractéristiques intrinsèques du matériel et du logiciel nous servant de base de travail. A ce titre, la sensibilité du récepteur, le temps nécessaire au traitement d'un paquet sans méthode d'accès et la qualité de la synchronisation par *beacons* ont été évalués de manière à connaître précisément les performances basiques du matériel utilisé pour la réalisation de notre réseau prototype. Dans un second temps, les fonctionnalités « SGTS » et « synchronisation par *cascades de beacons* » ont été validés. Une évaluation



du débit garanti par la méthode d'accès a aussi été réalisée. La partie prototypage a elle aussi donné des résultats encourageants : la synchronisation sur plusieurs niveaux de beacons permet de garder une synchronisation très correcte et la possibilité de pratiquer des accès simultanés au médium sans risque de collisions a été prouvée. L'*effet de capture* a pu être mis en évidence et les conditions nécessaires à l'attribution d'un SGTS ont été déterminées.

Le réseau prototype est à ce jour opérationnel et la méthode d'accès déterministe offre les propriétés attendues et des performances intéressantes. Le réseau est capable de fournir des garanties sur le débit et la latence au niveau de la méthode d'accès. Concrètement, en fixant les paramètres temporels intrinsèques du réseau, l'utilisateur peut rapidement, à partir des abaques obtenus et présentés dans le chapitre 4, déterminer le débit minimum et la latence maximum qu'il est en mesure d'attendre de la méthode d'accès.

## Perspectives

A partir des résultats obtenus dans ce travail, nous dégagons des perspectives de recherches nombreuses et variées. Certaines doivent être considérées à court terme et d'autres permettront de s'engager sur des travaux de recherche plus ambitieux à long terme.

Comme nous l'avons souvent évoqué précédemment, un point important serait de pouvoir assurer la transversalité de la gestion de la Qualité de Service au travers de toute la pile protocolaire, et plus particulièrement avec la couche directement supérieure, la couche *réseau*. Un jeu de primitives devra alors être créé et pris en charge par la ou les couches supérieures, y compris par un protocole de routage incluant une gestion de la QoS. Les paramètres du réseau pourraient alors être fixés automatiquement par ce processus en fonction des besoins de l'applicatif vis-à-vis des nœuds du réseau. Enfin, la couche MAC, en tant que premier témoin de l'activité du médium, pourrait remonter des informations sur son utilisation et sur les ressources libres. Ces informations pourraient être collectées aussi bien sur le supercoordonateur que sur les coordonneurs ou les nœuds terminaux du réseau, permettant ainsi au protocole de routage d'ajuster finement ses directives.

Au niveau de la topologie et du protocole de gestion de l'accès au médium, il serait intéressant de pouvoir étendre notre réseau sur une zone géographique plus importante ; la méthode d'accès proposée repose sur l'hypothèse que tous les coordonneurs sont à portée radio du supercoordonateur ; dans cette hypothèse, la synchronisation d'un coordonneur est directe et le réseau ne possède que deux niveaux de liens (supercoordonateur-coordonateur et coordonneur-nœud). Il serait donc intéressant de pouvoir mettre en œuvre des topologies moins restrictives, comme par exemple rattacher un coordonneur à un autre coordonneur si le premier n'est pas à portée du supercoordonateur. Une structure généralisable à  $n$  niveaux serait alors envisageable, plus flexible et plus apte à résoudre des problèmes réels. Nous pourrions facilement permettre cette extension en utilisant la négociation de SGTS, support simple et opérationnel pour permettre la négociation de slots pour des nœuds très éloignés. Reste cependant à régler le problème de la synchronisation qui, avec le protocole très simple utilisé actuellement, se dégrade fortement à chaque saut (d'après les résultats issus du prototypage). La conception d'un système de synchronisation plus précis, par exemple par adaptation d'un protocole existant comme NTP (*Network Time Protocol*) pourrait permettre de réduire considérablement la gigue de synchronisation sur le réseau global.

Concernant la phase de création du réseau, nous avons vu que le mécanisme d'association déterministe nécessite que le supercoordonateur soit le premier nœud éveillé. Dans l'état actuel de la proposition, un coordonneur qui se réveille écoute le médium indéfiniment jusqu'à recevoir un *superbeacon*, et donc, doit attendre le réveil du supercoordonateur pour commencer à diffuser ses propres *beacons* et contacter ses propres nœuds terminaux. Une perspective intéressante serait de prévoir un mécanisme permettant l'élection du supercoordonateur, si toutefois l'application le permet. Bien entendu, cette élection ne devrait en aucun cas affaiblir le côté déterministe de la phase de création du réseau.

Concernant la fiabilisation de la couche physique, une perspective intéressante serait la mise en œuvre

de codes correcteurs en dessous de la couche MAC (codes de *Hamming*, codes convolutionnels, *turbocodes*, *Reed Salomon*, FEC, etc.). Cette modification est tout à fait envisageable sur nos nœuds prototypes car le calcul du CRC peut être désactivé sur les modules 802.15.4 que nous utilisons. Ainsi, il est possible d'accéder aux données reçues sans tenir compte du CRC, ce qui rend possible cette perspective sur le plan pratique.

Au niveau des travaux de simulation, il serait intéressant de pouvoir implémenter la méthode d'accès dans un simulateur proposant des fonctionnalités de modélisation de la couche physique. Les résultats de cette opération seraient très intéressants pour entrevoir la capacité du réseau à *passer à l'échelle* – opération a priori très délicate à réaliser par prototypage. De plus, en reprenant la perspective de vaste étendue du réseau, l'étude par simulation constituerait, là aussi, un bon moyen de fixer les portées de chaque nœud pour un réseau étendu sur une grande zone géographique. Dans un premier temps, un modèle basé sur les *extensions sans fil* du logiciel NS2 serait envisageable, incluant un protocole de gestion fine de la synchronisation.

Concernant l'aspect prototypage, des mesures sur le délai engendré par l'association déterministe des nœuds au réseau pourraient être réalisées pour confronter les résultats obtenus par simulation. Des mesures concernant le délai d'acheminement d'un message de données en fonction du nombre de sauts devront également faire l'objet d'une campagne de mesures, les performances temporelles n'ayant été établies que par simulation et pour la phase particulière et originale de l'*association déterministe au réseau*. D'une manière générale, il serait bon de pouvoir comparer, quand cela est possible, les différentes méthodes de validation mises en œuvre durant la thèse (Réseaux de Pétri, simulation et prototypage). Par exemple, le modèle Réseaux de Pétri développé pourrait être amélioré et se voir ajouter une fonctionnalité temporelle par utilisation des *Réseaux de Pétri Temporisés* (*Timed Petri Nets*) ; les résultats obtenus par ce nouveau modèle plus complet pourraient alors être comparés directement aux résultats issus des travaux de simulation.

L'aspect énergétique pourrait lui aussi faire l'objet d'une analyse intéressante. Dans le cadre de notre réseau prototype, nous n'avons pas pu mettre en œuvre la fonctionnalité de mise en veille rapide des nœuds terminaux. Cet aspect pourrait donc aussi être étudié, dans un premier temps par simulation grâce à une modification du simulateur existant pour la prise en charge de l'aspect énergétique, comme cela a été fait pour l'aspect temporel. Dans un deuxième temps, cette étude pourrait être poursuivie par l'implémentation de la fonction de mise en veille rapide en mode *suivi de beacon* sur le réseau prototype. Une campagne de mesures par prototypage réel pourrait alors être réalisée et les résultats pourraient être comparés à ceux obtenus par simulation.

Enfin, une perspective intéressante serait la mise en œuvre de la méthode d'accès au médium déterministe sur la plateforme de robots présentée dans le chapitre 3. Cette plateforme est actuellement en cours d'élaboration et pourrait constituer une application de test concrète et réelle.

De toutes les perspectives évoquées ci-dessus, plusieurs font déjà l'objet de travaux de thèse au laboratoire. D'autres sont maintenant liées à un projet de recherche national auquel nous participons. Pour les perspectives restantes, nous espérons que cette thèse offrira une source d'inspiration à de futurs travaux doctoraux auxquels nous aimerions vivement contribuer.

# Bibliographie générale

- [APOS 96] C. APOSTOLAS, R. TAFAZOLLI et B.G. EVANS – *Comparison between elimination yield non pre-emptive priority multiple access (EY-NPMA) and dynamic TDMA (D-TDMA)* – 7th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRCap96) vol 2, issue 15-18 Oct 1996 pp 663-667 (1996)
- [ARCE] Le site web de l'ARCEP (Autorité de Régulation des Communications Electroniques et des Postes) – <http://www.arcep.fr>
- [ATAM 94] Y. ATAMNA – *Réseaux de Petri temporisés stochastiques et bien formés : définition, analyse et application aux systèmes distribués temps réel* – Doctorat de l'Université Paul Sabatier, Toulouse, France (1994)
- [BADI 04] H. BADIS et K. AL AGHA – *QOLSR Multi-path Routing for Mobile Ad Hoc Networks Based on Multiple Metrics* – 59th IEEE Vehicular Technology Conference (IEEE VTC'04-Spring), Milan, Italie (Mai 2004)
- [BERT 04] B. BERTHOMIEU, P.O. RIBET et F. VERNADAT – *The tool TINA - Construction of Abstract State Spaces for Petri Nets and Time Petri Nets* – International Journal of Production Research, Vol. 42, no 14, (Juillet 2004)
- [BETH 07] J. BETHMONT et P. POMMIER – *Réalisation d'une passerelle IP/ZigBee* – Rapport de projet tuteuré, équipe ICARE, Blagnac (Mars 2007)
- [BOUY 97] E. BOUYER et E. HORLAIT – *Bandwidth Management and Reservation over Shared Media – Segundo Seminário Franco-Brasileiro em Sistemas Informáticos Distribuídos (SFBSID'97)*, Fortaleza, Brazil (1997)
- [BOUZ 03] A. BOUZOUALEGH, T. VAL, F. PEYRARD et E. CAMPO – *Study of the Characteristics Needed for Underwater Acoustic Networks* – 7th WSEAS International Conference on Circuits, Systems, Communications and Computers (CSCC'2003), Corfu, Grece (Juillet 2003)
- [BRAM 83] G.W. BRAMS – *Réseaux de Petri : théorie et pratique* – Masson, Paris (1983)
- [CALL 02] E. CALLAWAY, P. GORDAY, L. HESTER, J.A. GUTIERREZ, M. NAEVE, B. HEILE et V. BAHL – *Home Networking with IEEE 802.15.4 : a developing standard for low-rate wireless personal area networks* – IEEE Communications Magazine vol. 40, no. 8, pp 70-77 (Août 2002)
- [CAPP 02] Communauté d'agglomération Pau Pyrénées – PAU BROADBAND COUNTRY (*large bande pour tous*) – Dossier de présentation (Février 2002)
- [CHIC 05] O. CHICHEPORTICHE – *Airbus passe à la VoIP avec France Télécom* – <http://www.silicon.fr/fr/silicon/news/2005/07/12/airbus-passe-voip-france-telecom>
- [CHOI 03] S. CHOI, J. DEL PRADO, S. SHANKAR et S. MANGOLD – *IEEE 802.11e Contention-Based Channel Access (EDCF) Performance Evaluation* – IEEE International Conference on Communications (ICC'03) vol.2, pp 1151-1156 (Mai 2003)
- [CULL 04] D. CULLER, D. ESTRIN et M. SRIVASTAVA – *Overview of Sensor Networks* – IEEE Computer vol.37, issue 8 (Août 2004)
- [DESG 03] G. DESGEORGES – *Réseaux haut débit et réseaux sans fil* – Cours d'ingénieur ESEO, spécialité R&T (2003)
-

- [ELHO 04] S. EL HOMSI, E. CAMPO, T. VAL et J.J. MERCIER – *Utilisation de la technologie Bluetooth pour la transmission de données numériques dans les applications à contraintes temporelles variées* – Congrès des doctorants de l'Ecole Doctorale de Génie Electrique, Electronique et Télécommunications (GEET), Supaero, Toulouse (Mars 2004)
- [ETS1 06] ETSI PROJECT BROADBAND RADIO ACCESS NETWORKS – *Broadband Radio Access Networks (BRAN) : HiperMAN Physical (PHY) Layer* – ETSI Technical Specification ETSI TS 102 177 V1.3.1 (Février 2006)
- [ETS2 06] ETSI PROJECT BROADBAND RADIO ACCESS NETWORKS – *Broadband Radio Access Networks (BRAN) : HiperMAN Data Link Control (DLC) Layer* – ETSI Technical Specification ETSI TS 102 178 V1.3.2 (Mars 2006)
- [ETSI 00] ETSI PROJECT BROADBAND RADIO ACCESS NETWORKS – *Broadband Radio Access Networks (BRAN) : HiperLAN Type 2 System Overview* – ETSI Technical report ETSI TR 101 683 V1.1.1 (Février 2000)
- [ETSI] Le site web de l'ETSI (*European Telecommunications Standards Institute*) – <http://www.etsi.org>
- [EVOL] Le site web de EVOLUTION ROBOTICS – [http://www.evolution.com/er1/what\\_overview.masn](http://www.evolution.com/er1/what_overview.masn)
- [FLOR 85] G. FLORIN – *Réseaux de Petri stochastiques : théorie et méthode de calcul* – Doctorat de l'Université Pierre et Marie Curie Paris VI, Paris, France (1985)
- [FON] Le mouvement FON – <http://www.fon.com>
- [FRAN 06] J. FRANCOMME, G. MERCIER et T. VAL – *A simple method for guaranteed deadline of periodic messages in 802.15.4 cluster cells for automation control applications* – *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'2006)*, Prague. Czech Republic (Septembre 2006)
- [FRE1 04] FREESCALE SEMICONDUCTOR – *ZigBee / IEEE 802.15.4 FREESCALE solution* – Technology General Information (2004)
- [FRE2 04] FREESCALE SEMICONDUCTOR – *ZigBee Reference Design (ZRD01) documentation & applications notes* (2004)
- [FRE3 04] FREESCALE SEMICONDUCTOR – *MC13192 Reference Manual* (2004)
- [FRE4 04] FREESCALE SEMICONDUCTOR – *802.15.4 MAC/PHY Software User's Guide* (2004)
- [GANG 04] Gang LU, Bhaskar KRISHNAMACHARI et Cauligi S. RAGHAVENDRA – *Performance evaluation of the IEEE 802.15.4 MAC for Low-Rate Low-Power Wireless Networks* – Workshop on Energy-Efficient Wireless Communications and Networks (EWCN '04), IEEE International Performance Computing and Communications Conference (IPCCC), Phoenix, Arizona, USA (Avril 2004)
- [GARC 02] J.A. GARCÍA-MACÍAS, F. ROUSSEAU, G. BERGER-SABBATEL, L. TOUMI et A. DUDA – *Différenciation des services sur les réseaux sans-fil 802.11* – 9<sup>ieme</sup> Colloque Francophone sur l'Ingénierie des Protocoles (CFIP'2002), Montréal, Quebec (Mai 2002)
- [GERB 05] M. GERBE – *Réalisation d'un capteur sans fil communicant par ZigBee* – Rapport de stage, équipe ICARE, Blagnac (Juin 2005)
- [GOLD 01] M. GOLDHAMMER – *ETSI BRAN HiperMAN Liaison Report 2001-11-08*
- [GUIL 05] C. GUILLEMINOT, L. ANDRIEUX et J.J. MERCIER – *A contribution to a virtual prototyping for a spread spectrum telecommunication system using VHDL-AMS* – BMAS'05, San Jose, Californie, USA (Septembre 2005)
- [HOWE 87] D.A. HOWE – *Ku-band Satellite Two-Way Timing Using a Very Small Aperture Terminal (VSAT)* – 41st Annual Symposium on Frequency Control pp 149-160 (1987)
- [HOYM 01] C. HOYMAN, M. PÜTTNER et I. FORKEL – *The HiperMAN Standard - a Performance Analysis*
- [HPSI] Le site web du logiciel d'animation de RdP HPSIM, de H. ANSCHUETZ – <http://www.winpesim.de>
- [IEE1 02] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.15.1 IEEE Standard for Information technology, Part 15.1 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANS)* – IEEE Std 802.15.1-2002 (2002)

- 
- [IEE1 03] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.15.3 IEEE Standard for Information technology, specific requirements part 15.3 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for high rate wireless personal area networks (WPANs)* – IEEE Std 802.15.3-2003 (2003)
  - [IEE1 04] IEEE COMPUTER SOCIETY and IEEE MICROWAVE THEORY AND TECHNIQUES SOCIETY – *802.16 IEEE Standard for Local and metropolitan area networks – part 16 : Air Interface for Fixed Broadband Wireless Access Systems* – IEEE Std 802.16-2004 (Octobre 2004)
  - [IEE1 05] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.11 IEEE Standard for Information technology, Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Amendment : Medium Access Control (MAC) Quality of Service (QoS) Enhancements* – IEEE (2005)
  - [IEE1 99] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.11 IEEE Standard for Information technology, Specific Requirements – Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications* – IEEE standard 8802-11 (1999)
  - [IEE2 03] LAN-MAN Standards Committee of the IEEE COMPUTER SOCIETY – *802.15.4 IEEE Standard for Information technology, Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)* – IEEE Std 802.15.4-2003 (2003)
  - [IEE3 03] IEEE COMPUTER SOCIETY – *IEEE 802.15.4/D18, Draft Standard : Low Rate Wireless Personal Area Networks* – (Février 2003)
  - [IEEE] Le site web du comité de standardisation IEEE 802 (*IEEE 802 LAN/MAN Standards Committee*) – [http ://www.ieee802.org](http://www.ieee802.org)
  - [IRDA] Le site web de IrDA (*Infrared Data Association*) – [http ://irda.org](http://irda.org)
  - [KHOU 05] T. KHOUTAIF et F. PEYRARD – *Performances evaluation of the asynchronous Bluetooth links in a real time environment* – *IEEE Personal Wireless Communications (PWC'2005)*, Colmar, France (Août 2005)
  - [KHOU 06] T. KHOUTAIF et G. JUANOLE – *Formal modelling and evaluation of the data transfer phase of the ACL links on the WPAN Bluetooth* – 11th International Conference on Emerging Technologies and Factory Automation (ETFA'2006), Prague, République Tchèque (Septembre 2006)
  - [LAAS] Le site web du LAAS – [http ://www.laas.fr](http://www.laas.fr)
  - [LEPA 91] F. LEPAGE – *Les Réseaux Locaux Industriels - Principes illustrés par des exemples* – Hermes (1991)
  - [LLI1 06] J.F. LLIBRE, P. PINEL et E. CAMPO – *Dimensionnement d'un générateur photo voltaïque pour un système communicant autonome* – XII<sup>ième</sup> Colloque National de la Recherche dans les IUT, Brest, France (2006)
  - [LLI2 06] J.F. LLIBRE, P. PINEL et E. CAMPO – *Quel choix de source d'énergie pour rendre un système communicant autonome ?* – XII<sup>ième</sup> Colloque National de la Recherche dans les IUT, Brest, France (2006)
  - [MARS 85] M.A. MARSAN, G. BALBO, A. BOBBIO, G. CHIOLA, G. CONTE et A. CUMANI – *On Petri Nets with Stochastic Timing* – International Workshop on Timed Petri Nets (INTPN'85), Turin, Italie (Juillet 1985)
  - [MAUR 05] R. MAURICE – *Contribution à la méthodologie de conception système : application à la réalisation d'un microsystème multicapteurs communicant pour le génie civil* – Doctorat de l'Institut National Polytechnique de Toulouse, Rapport LAAS n°05646, Toulouse, France (2005)
  - [MAXS] Le site web de MAXSTREAM – [http ://www.maxstream.net](http://www.maxstream.net)
  - [MERC 03] A. MERCIER, P. MINET et L. GEORGES – *Introducing QoS support in Bluetooth Piconet with a Class-Based EDF Scheduling* – Rapport de recherche de l'INRIA RR-5054 (Décembre 2003)
  - [MERC 04] G. MERCIER – *Un réseau sans fil faible consommation : IEEE 802.15.4 (alias ZigBee)* (2004)
  - [MERC 06] J.J. MERCIER – *Computers II : Séquence après séquence, Introduction aux Graphts* – chapitre 3.2, Collection Technosup, Ellipses (2006)
-

- [MERL 76] P.M. MERLIN – *Recoverability of communication protocols, implication of a theoretical study* – IEEE Transaction on Communications, 24(9) :1036-1043 (1976)
- [MICH 05] F. MICHAUT et F. LEPAGE – *Adaptation des applications distribuées à la qualité de service du réseau de communication – Ecole d'été temps réel 2005* (ETR'05) Nancy, France (Septembre 2005)
- [MUHL 02] P. MUHLETHALER – *802.11 et les réseaux sans fil* – Eyrolles (2002)
- [NIC1 98] K. NICHOLS, S. BLAKE, F. BAKER et D. BLACK – *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers* – RFC 2474, IETF Network Working Group (Decembre 1998)
- [NIC2 98] K. NICHOLS, S. BLAKE, F. BAKER et D. BLACK – *An Architecture for Differentiated Service* – RFC 2475, IETF Network Working Group (Decembre 1998)
- [NS] Le site web de *Network Simulator* (NS) – <http://www.isi.edu/nsnam/ns>
- [OLC] Le site web du groupe OLC – <http://www2.laas.fr/laas/1-4286-Groupe-OLC.php>
- [OPNE] Le site web d'*OPNET* – <http://www.opnet.com>
- [PERK 97] C. PERKINS et E. ROYER – *Ad-hoc on-demand distance vector routing* – MILCOM'97 (Novembre 1997)
- [PETR 06] M. PETROVA, J. RIIHIJARVI, P. MAHONEN et S. LABELLA – *Performance Study of IEEE 802.15.4 Using Measurements and Simulations – IEEE Wireless Communications and Networking Conference* (WCNC'06), vol. 1, pp : 487- 492, Las Vegas, Nevada, USA (Avril 2006)
- [PUJO 95] G. PUJOLLE – *Les réseaux* – Eyrolles (1995)
- [QUAL 02] SCALABLE NETWORK TECHNOLOGIES – *Speed & Scalability : Requirements for Industrial Strength Network Simulators* – QUALNET White Paper (2002)
- [QUAL] Le site web de *Qualnet* – <http://www.scalable-networks.com>
- [RAN] Le *web ring* des RAN (*Rural Area Network*) – <http://ran.vaour.net/webring>
- [SAMS] SAMSUNG/CUNY – *NS2 simulator for 802.15.4* – <http://www-ee.ccny.cuny.edu/zheng/pub>
- [SIMP 05] D. SIMPLOT-RYL – *Real-time aspects in Wireless Sensor Networks* – Ecole d'Eté Temps Réel 2005 (ETR'05), Nancy, France (Septembre 2005)
- [SMER 04] S.A. SMERZI, G. GIRLANDO, T. COPANI et G. PALMISANO – *A Ku-band monolithic receiver for DVB-S applications* – IEEE Communications Magazine, vol.42 issue 8, pp 132-139 (Août 2004)
- [THOM 05] J.P. THOMESSE – *Fieldbus Technology and Industrial Automation – 10th IEEE Conference on Emerging Technologies and Factory Automation* (ETFA 2005), Catania, Italie (Septembre 2005)
- [TINA] Le site web de TINA – <http://www.laas.fr/tina>
- [TSF] Le site web de l'association TOULOUSE SANS FIL – <http://www.toulouse-sans-fil.net>
- [VAL 05] T. VAL et G. JUANOLE – *La Qualité de Service dans les réseaux sans fil* – École d'été Temps Réel (ETR'05), Nancy, France (Septembre 2005)
- [VAND 01] M. VAN DER ZEE et G. HEIJENK – *Quality of Service in Bluetooth Networking* – 10/0362-FCP NB 102 88 Uen (Janvier 2001)
- [VDB 06] A. VAN DEN BOSSCHE, T. VAL et E. CAMPO – *Proposition of a full deterministic medium access method for wireless network in a robotic application – 63rd IEEE Vehicular Technology Conference* (VTC-2006-Spring), Melbourne, Australie (Mai 2006)
- [VDB 07] A. VAN DEN BOSSCHE, T. VAL et E. CAMPO – *Une méthode d'accès totalement déterministe pour un réseau personnel sans fil* – 8<sup>ième</sup> congrès de l'Ecole Doctorale Systèmes de Toulouse (EDSYS'07), Albi, France (Mai 2007)
- [VDB1 05] A. VAN DEN BOSSCHE, T. VAL et E. CAMPO – *Métrologie pour l'analyse comparative des performances temporelles des liens Bluetooth – IEEE Sciences of Electronic, Technologies of Information and Telecommunication* (SETIT'2005), Sousse, Tunisie (Mars 2005)
- [VDB2 05] A. VAN DEN BOSSCHE et M. GERBE – *Les modules IEEE 802.15.4 ICARE* – Documentation technique illustrée, document interne ICARE (Juin 2005)

- 
- [VDB3 05] A. VAN DEN BOSSCHE – *IEEE 802.15.4 / ZigBee : un réseau sans fil à faible consommation énergétique* – 2<sup>èmes</sup> rencontres des Sciences et Technologies de l'Information (ASTI'2005), Aubière, France (Octobre 2005)
- [VINT 95] *The NS Manual (formerly NS Notes and Documentation)* par les contributeurs du projet VINT – UC Berkeley, LBL, USC/ISI, and Xerox PARC (1995)
- [WEI 02] A. WEI et S. BOUMERDASSI – *Support de la QoS dans les réseaux 802.11* – Rapport scientifique CEDRIC N° 401 (2002)
- [WEIN 97] J. WEINMILLER, M. SCHLÄGER, A. FESTAG et A. WOLISZ – *Performance Study of Access Control in Wireless LANs - IEEE 802.11 DFWMAC and ETSI RES 10 HiperLAN – Mobile Networks and Applications*, vol.2, pp 55-67 (1997)
- [WROC 97] J. WROCLAWSKI – *The Use of RSVP with IETF Integrated Services* – RFC 2210, IETF Network Working Group (Septembre 1997)
- [XU 02] K. XU, M. GERLA et S. BAE – *How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks* – *IEEE Global Telecommunications Conference (GLOBECOM'02)* vol.1, pp 72-76 (Novembre 2002)
- [YU 04] J. YU – *IEEE 802.11e QoS for Wireless LAN : A Research Direction* – *Multimedia Networking Research Laboratory (MNLAB) Networking Seminar* (Décembre 2003)
- [ZBAL 05] ZIGBEE ALLIANCE – *ZigBee Specification* – ZigBee Document 053474r06, version 1.0 (2005)
- [ZBAL] Le site web de la ZIGBEE ALLIANCE – [www.zigbee.org](http://www.zigbee.org)
- [ZHEN 04] J. ZHENG et M. J. LEE – *Will IEEE 802.15.4 make ubiquitous networking a reality ? : A discussion on a potential low power, low bit rate standard* – *IEEE Communications Magazine*, vol. 27, no. 6, pp. 23-29 (2004)
- [ZHEN 06] J. ZHENG et M. J. LEE – *A comprehensive performance study of IEEE 802.15.4* – *Sensor Network Operations*, IEEE Press, Wiley Interscience, chapitre 4, pp. 218-237 (2006).





# Glossaire

- 16QAM, *16-ary Quadrature Amplitude Modulation*
- 64QAM, *64-ary Quadrature Amplitude Modulation*
- AES, *Advanced Encryption Standard*
- AFH, *Adaptive Frequency Hopping* : Technique de saut de fréquence évitant les canaux encombrés
- AIFS, *Arbitration Inter-Frame Spacing*
- AODV, *Ad hoc On Demand Distance Vector*
- ARCEP, Autorité de Régulation des Communications Electroniques et des Postes
- ART, Autorité de Régulation des Télécommunications
- ASP, *Application Support Package*
- ATM, *Asynchronous Transfer Mode*
- ARQ, *Automatic Repeat reQuest*
- BDM, *Background Debug Mode*
- BER, *Bit Error Rate* : taux d'erreur bit
- BI, *Beacon Interval*
- BLR, Boucle Locale Radio
- BO, *Beacon Order*
- BPSK, *Binary Phase Shift Keying*
- BSN, *Beacon Sequence Number*
- CAP, *Contention Access Period*
- CCA, *Clear Channel Assessment*
- CCK, *Complementary Code Keying*
- CDMA, *Code Division Multiple Access*
- CFP, *Contention Free Period*
- CSMA/CA, *Carrier Sense Multiple Access with Collision Avoidance*
- CSMA/CD, *Carrier Sense Multiple Access with Carrier Detect*
- CTS, *Clear To Send*
- CTS, *Clear To Send* : Prêt à recevoir l'envoi. Terminologie empruntée à RS232. Voir aussi RTS.
- DECT *Digital Enhanced Cordless Telephone*
- DIFS, *Distributed Inter-Frame Spacing*
- DSL, *Digital Subscriber Line*
- DSN, *Data Sequence Number*
- DSSS, *Direct Sequence Spread Spectrum*
- DVB, *Digital Video Broadcasting*
- DVB-S, *Digital Video Broadcasting - Satellite*
- DVB-T, *Digital Video Broadcasting - Terrestrial*
- EDGE, *Enhanced Data Rates for GSM Evolution*
- ED, *Energy Detection*
- EY-NPMA, *Elimination Yield – Non-preemptive Priority Multiple Access*
- FCS, *Frame Check Sequence*
- FDD, *Frequency Division Duplexing*
- FDMA, *Frequency Division Multiple Access*
- FEC, *Forward Error Correction*
- FFD, *Full Function Device*

- FHSS, *Frequency Hopping Spread Spectrum*
- FSK, *Frequency Shift Keying*
- FTTH, *Fiber To The Home*
- GBS, *Guaranteed Beacon Slot*
- GFSK, *Gaussian Frequency Shift Keying*
- GMSK, *Gaussian Minimum Shift Keying*
- GPIO, *General Purpose Input/Output*
- GPRS, *General Packet Radio Service*
- GTS, *Guaranteed Time Slot*
- IEEE, *Institute of Electrical and Electronics Engineers*
- IP, *Internet Protocol*
- ISO, *International Organization for Standardization*
- L2CAP, *Logical Link Control and Adaptation Layer Protocol*
- LAN, *Local Area Network* : Réseau local
- LATTIS, *LABoratoire Toulousain des Technologies et de l'Ingénierie des Systèmes*
- LIFS, *Long Inter-Frame Spacing*
- LLC, *Logical Link Control*
- LMDS, *Local Multipoint Distribution Service*
- LP-WPAN, *Low Power – Wireless Personal Area Network*
- LQI, *Link Quality Indication*
- MAC, *Medium Access Control*
- MAN, *Metropolitan Area Network* : Réseau métropolitain
- MCPS, *MAC Common Part Sublayer*
- MCPS-SAP, *MAC Common Part Sublayer – Service Access Point*
- MIMO, *Multiple Input/Multiple Output*
- MLME-SAP, *Network Layer Management Entity – Service Access Point*
- MLME, *Network Layer Management Entity*
- MPDU, *MAC-level Protocol Data Unit* : unité de données du protocole de niveau Liaison
- NAV, *Network Allocation Vector*
- NHLE, *Next Higher Level Entity*
- NIB, *Network-layer Information Base*
- NLDE-SAP, *Network Layer Data Entity – Service Access Point*
- NLDE, *Network Layer Data Entity*
- NLME, *Network Layer Management Entity*
- NPDU, *NWK-level Protocol Data Unit* : unité de données du protocole de niveau Réseau
- OFDM, *Orthogonal Frequency Division Multiplexing*
- OLSR, *Optimized Link State Routing Protocol*
- O-QPSK, *Orthogonal-Quadrature Phase Shift Keying*
- Overhead : surplus de données non-utiles pourtant nécessaires
- PAN, *Personal Area Network* : Réseau personnel
- PDS, *Previously Dedicated Slot*
- PDU, *Protocol Data Unit* : unité de données du protocole
- PIFS, *Point Coordination Inter-Frame Spacing*
- PN-code, *Pseudo Noise code* : code pseudo aléatoire
- PPDU, *PHY-level Protocol Data Unit* : unité de données du protocole de niveau Physique
- QoS, *Quality of Service*
- QPSK, *Quadrature Phase Shift Keying*
- RFD, *Reduced Function Device*
- RTS, *Request To Send* : Demande d'envoi. Terminologie empruntée à RS232. Voir aussi CTS.
- SAP, *Service Access Point* : point d'accès de service
- SDU, *Service Data Unit* : unité de données de service
- SIFS, *Short Inter-Frame Spacing*
- SNR, *Signal-to-Noise Ratio* : rapport signal sur bruit
- SO, *Superframe Order*
- SGTS, *Simultaneous Guaranteed Time Slot*
- SSCS, *Service Specific Convergence Sublayer* (IEEE 802.15.4)

- TDD, *Time Division Duplexing*
- TDMA, *Frequency Division Multiple Access*
- TKIP, *Temporal Key Integrity Protocol*
- UMTS, *Universal Mobile Telecommunications System*
- VSAT, *Very Small Aperture Terminal*
- WAN, *Wide Area Network* : Réseau étendu
- WEP, *Wired Equivallent Privacy*
- WLAN, *Wireless Local Area Network* : Réseau local sans fil
- WMAN, *Wireless Metropolitan Area Network* : Réseau métropolitain sans fil
- WPA2, *Wi-Fi Protected Access*
- WPAN, *Wireless Personal Area Network* : Réseau personnel sans fil
- WWAN, *Wireless Wide Area Network* : Réseau étendu sans fil
- xDSL, voir DSL
- ZC, *ZigBee Coordinator*
- ZDO, *ZigBee Device Object*
- ZED, *ZigBee End Device*
- ZR, *ZigBee Router*



# Table des figures

1.1	Différents types de réseaux sans fil et portées typiques . . . . .	13
1.2	Topologie <i>Infrastructure</i> de 802.11 . . . . .	19
1.3	Topologie <i>Ad-hoc</i> de 802.11 . . . . .	19
1.4	Topologie point à point de Bluetooth . . . . .	23
1.5	Topologie <i>Piconet</i> de Bluetooth . . . . .	23
1.6	Topologie <i>Scatternet</i> de Bluetooth . . . . .	23
1.7	Le principe de l' <i>Ultra Wide Band</i> : utilisation du spectre radio fréquence . . . . .	24
1.8	Illustration du protocole RSVP . . . . .	31
1.9	Procédé de multiplexage / démultiplexage pour la cohabitation de plusieurs trafics sur le même canal de communication . . . . .	33
1.10	Illustration de CSMA/CA si le médium est initialement libre . . . . .	35
1.11	Illustration de CSMA/CA si le médium est initialement occupé . . . . .	35
1.12	Topologie présentant deux terminaux mutuellement cachés . . . . .	36
1.13	Déroulement de l'accès au médium par la méthode CSMA/CA . . . . .	36
1.14	Déroulement d'un accès au médium réussi par la méthode EY-NPMA . . . . .	38
1.15	Déroulement d'un accès au médium par la méthode EY-NPMA ayant échoué . . . . .	38
1.16	La supertrame HiperLAN/2 . . . . .	39
1.17	Déroulement temporel d'une période d'accès sans contention dans le cadre de 802.11 PCF . . . . .	41
2.1	Graphe des états d'un modem 802.15.4 (source FREESCALE) . . . . .	54
2.2	Représentation de la topologie en étoile . . . . .	55
2.3	Représentation de la topologie point à point . . . . .	55
2.4	Le principe générique de l'encapsulation . . . . .	58
2.5	Communication respectant le protocole normalisé . . . . .	58

---

2.6	La pile protocolaire 802.15.4 / ZigBee . . . . .	59
2.7	Principe d'interfaçage entre couches et SAP pour ZigBee . . . . .	60
2.8	Rapport entre BER et SNR pour diverses technologies de transmission sans fil (source IEEE) . . . . .	62
2.9	Structure du paquet de niveau physique . . . . .	63
2.10	Principe du transfert de données dans une étoile . . . . .	64
2.11	Représentation d'une supertrame IEEE 802.15.4 . . . . .	65
2.12	Structure de la trame balise . . . . .	68
2.13	Structure de la trame de données . . . . .	69
2.14	Structure de la trame d'acquiescement . . . . .	69
2.15	Structure de la trame de service . . . . .	69
2.16	Exemple de création du réseau ZigBee en arbre . . . . .	71
2.17	Exemple de topologie maillée . . . . .	72
2.18	Structure de la couche réseau proposée par ZigBee . . . . .	72
2.19	Illustration de la hiérarchie des adresses dans le cadre de l'adressage automatique en arbre	74
2.20	Structure du paquet de niveau <i>réseau</i> et encapsulation dans une trame de données 802.15.4	76
2.21	Les messages <b>GTS.request</b> peuvent entrer en collision avec le reste du trafic CSMA/CA	77
2.22	Le service garanti (GTS) ne peut être maintenu de manière déterministe . . . . .	78
2.23	Un GTS inutilisé constitue une perte de bande passante non réutilisable . . . . .	79
2.24	Plusieurs coordinateurs attribuant des GTS peuvent provoquer des <i>collisions de GTS</i> .	79
3.1	Topologie proposée pour la communication interne au robot . . . . .	88
3.2	Topologie proposée pour la communication entre robots . . . . .	88
3.3	Éléments du réseau, liens et portées minimales pour le bon fonctionnement du protocole	93
3.4	Répartition des beacons au sein de la supertrame . . . . .	94
3.5	Périodes des cycles déterministes en fonction du niveau de réservation du PDS . . . . .	95
3.6	Topologie du réseau mis en place pour l'exemple . . . . .	97
3.7	Un exemple de déroulement du protocole . . . . .	98
3.8	Topologie d'école du réseau illustrant la négociation d'un SGTS . . . . .	101
3.9	Principe de la négociation protocolaire d'un SGTS . . . . .	102
3.10	Topologie du réseau pour la négociation d'un SGTS . . . . .	102
3.11	Principe de la négociation d'un SGTS dans le cadre de notre topologie . . . . .	103

4.1	Séquencement du protocole pour une demande d'association d'un nœud au réseau . . .	113
4.2	Éléments du réseau et liens entre ces éléments . . . . .	114
4.3	Représentation du modèle Réseau de Petri . . . . .	115
4.4	Exemple d'une répartition de PDS/GBS/GTS sous forme matricielle . . . . .	120
4.5	Représentation des fenêtres temporelles pour l'association déterministe d'un coordinateur (pour $BO = 3$ ) . . . . .	121
4.6	Représentation des fenêtres temporelles pour l'association déterministe d'un nœud termi- nal (pour $BO = 3$ et $n_{GBS} = 0$ ) . . . . .	123
4.7	Représentation des fenêtres temporelles pour l'association déterministe d'un nœud termi- nal (pour $BO = 3$ et $n_{GBS} = 1$ ) . . . . .	124
4.8	Débit utile avec CSMA/CA dans un réseau 802.15.4 avec <i>beacons</i> . . . . .	126
4.9	Quelques plateformes IEEE 802.15.4/ZigBee de FREESCALE . . . . .	128
4.10	Utilisation conjointe du module espion et de l'analyseur de trames . . . . .	130
4.11	Évaluation de la proportion de trames perdues en fonction de la puissance du signal reçu	131
4.12	Illustration des délais engendrés sur toute la transmission avec SMAC et le mode <i>packet</i>	132
4.13	Mesure du délai total (temps de traitement et de transmission) en fonction de la longueur utile du paquet transmis (PSDU) . . . . .	133
4.14	Évaluation du débit utile maximal en fonction de la longueur du paquet transmis . . . .	134
4.15	Topologie pour la mesure de la gigue de synchronisation pour deux nœuds d'une même étoile . . . . .	135
4.16	Quantification de la gigue de synchronisation pour deux nœuds d'une même étoile . . .	135
4.17	Topologie pour la mesure de la gigue de synchronisation pour deux nœuds de deux étoiles différentes . . . . .	136
4.18	Quantification de la gigue de synchronisation pour deux nœuds de deux étoiles différentes	137
4.19	Évaluation du débit utile maximal pour un slot déterministe, en fonction du niveau de $BO$ et du niveau de réservation $n$ . . . . .	138
4.20	Évaluation du débit maximum offert en fonction de la charge globale du réseau . . . .	139
4.21	Succession des messages échangés sur le médium dans le protocole de test pour la faisa- bilité des SGTS . . . . .	140
4.22	Taux de trames réceptionnées correctement en fonction du rapport de puissances reçues	141
4.23	Taux de trames réceptionnées correctement en fonction du rapport de puissances reçues	142
4.24	Le robot ER1, le capteur infrarouge et le coordinateur (module ZigBee ICARE) . . . .	143
4.25	Réalisations de modules ZigBee ICARE . . . . .	143
4.26	Passerelle IP/ZigBee . . . . .	144

4.27 Les deux robots suiveurs de ligne et le réseau formé . . . . .	145
---	-----



# Liste des tableaux

2.1	Consommation typique d'un modem 802.15.4 avec temps de basculement entre chaque état (source FREESCALE) . . . . .	54
2.2	Caractéristiques des deux couches physiques proposées (PHY868/915 et PHY2450) . . .	61
2.3	Structure de la table de routage d'un nœud ZigBee . . . . .	75
4.1	Éléments validés par chaque méthode utilisée . . . . .	111

---

## **Proposition d'une nouvelle méthode d'accès déterministe pour un réseau personnel sans fil à fortes contraintes temporelles**

Aujourd'hui, les technologies de réseaux sans fil (WLAN/WPAN) présentes sur le marché sont nombreuses et globalement complémentaires. Cependant, trop peu d'entre elles proposent de réelles garanties sur la remise de messages dans un temps borné alors que ces exigences sont fondamentales dans le cadre d'une utilisation de type industriel. Dans le cadre de nos travaux, nous proposons une nouvelle couche MAC *entièrement déterministe* pour un réseau sans fil personnel basse consommation (LP-WPAN) IEEE 802.15.4 présentant des garanties sur le plan temporel.

Tout d'abord, un état de l'art est effectué sur les principaux réseaux sans fil existants, les mécanismes de gestion de la Qualité de Service et les méthodes d'accès généralement utilisées. Nous présentons ensuite la technologie IEEE 802.15.4/ZigBee sur laquelle sont basés nos travaux. Cette étude approfondie nous a permis d'identifier certaines imperfections au niveau de la méthode d'accès par rapport aux contraintes temporelles. Nous proposons de combler ces lacunes par la création d'une couche MAC entièrement déterministe dont nous présentons les caractéristiques et les nouvelles fonctionnalités. Plusieurs méthodes complémentaires ont été utilisées pour valider nos propositions : Réseaux de Petri, simulation et prototypage réel. Les résultats obtenus et les analyses de ces trois études sont exposés.

---

## **Proposition of a new determinist medium access method for a wireless personal area network with hard temporal constraints**

Today, wireless network technologies (WLAN/WPAN) available on the market are numerous and complementary. Nevertheless, none of these technologies present some real guaranties on message delivery within a bounded delay. However, this kind of constraint is a recurrent need in the industrial communication application context. To solve this problem, our research work proposes an original full deterministic MAC layer for an IEEE 802.15.4 Low-Power Wireless Personal Area Network.

We first present a state of the art on principal wireless network standards and technologies, Quality of Service protocols and commonly used Medium Access Control mechanisms. Then, we present the IEEE 802.15.4/ZigBee wireless technology which is used as a basis for our works. This study has enabled us to identify weakness on 802.15.4 Medium Access Control, in particular for highly time-constrained data transports. Our research works propose to solve these identified weaknesses by proposing an original full deterministic medium access method. Several validation methods/tools were used: Petrinets, simulation and prototyping. Obtained results are then exposed and discussed.

---